

NOTE: THIS BUG. S/B
ENHANCED, AS PER,
EG. 6, 055, 295

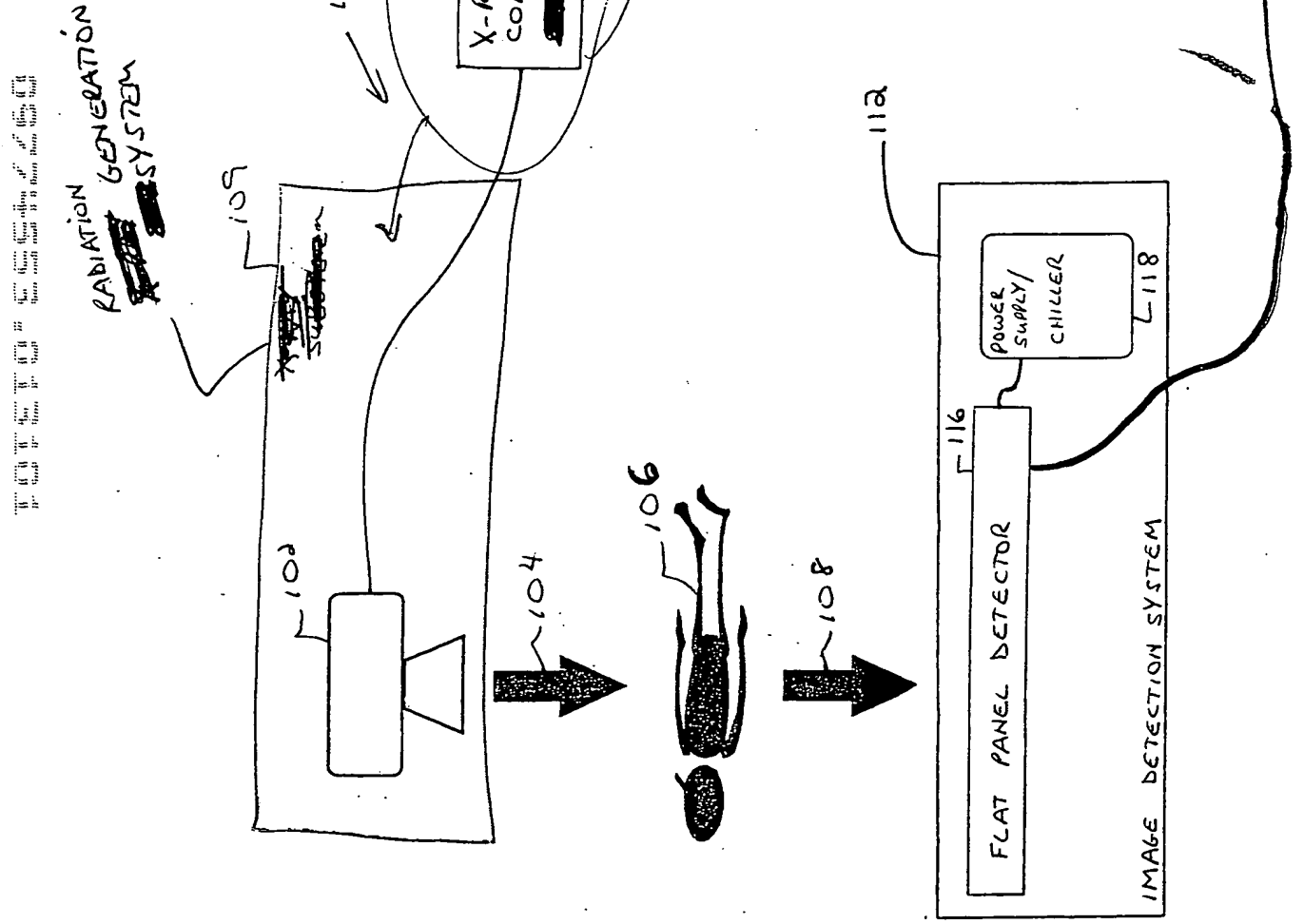


FIG. 1

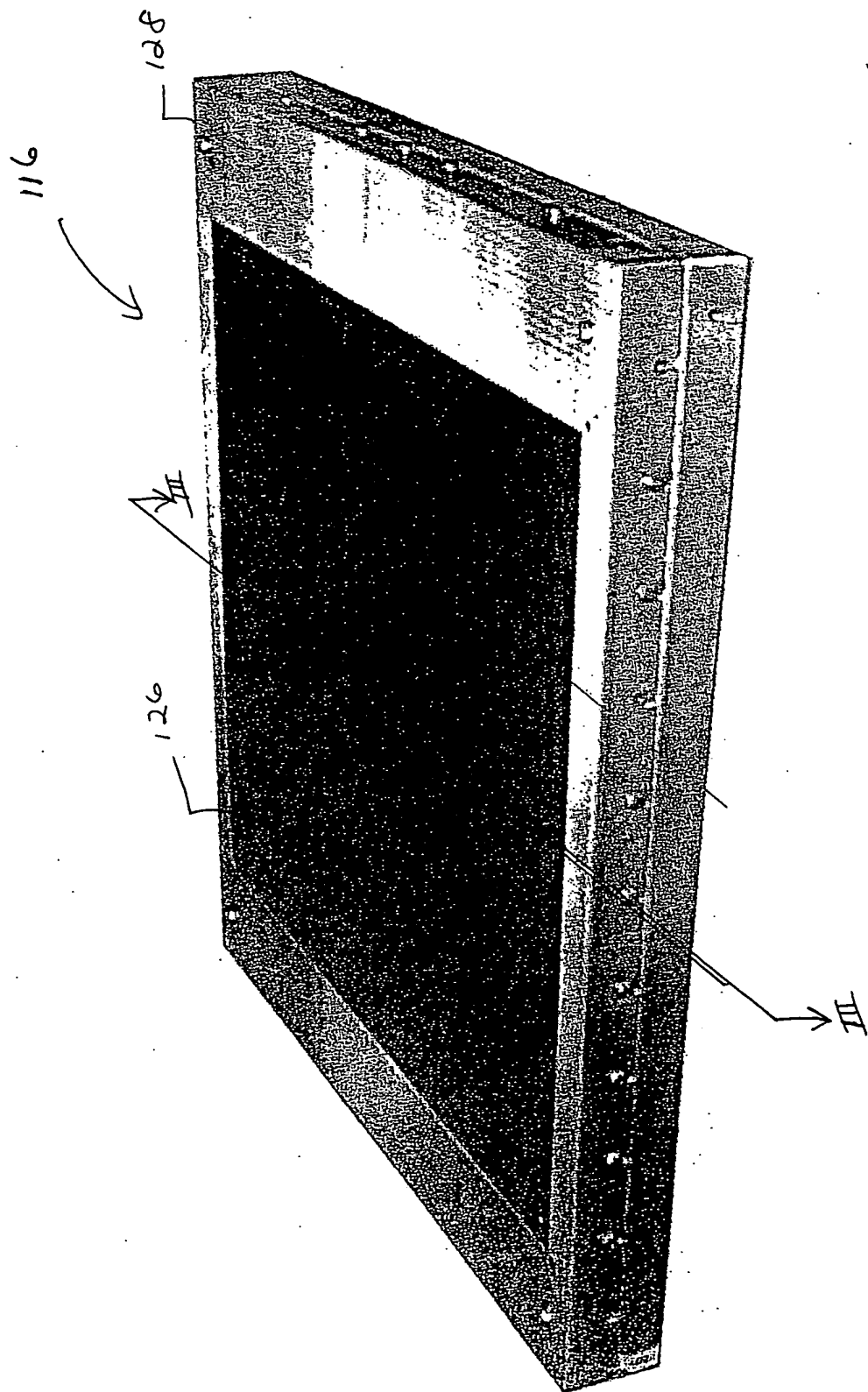


FIG. 2 (PRIOR ART)

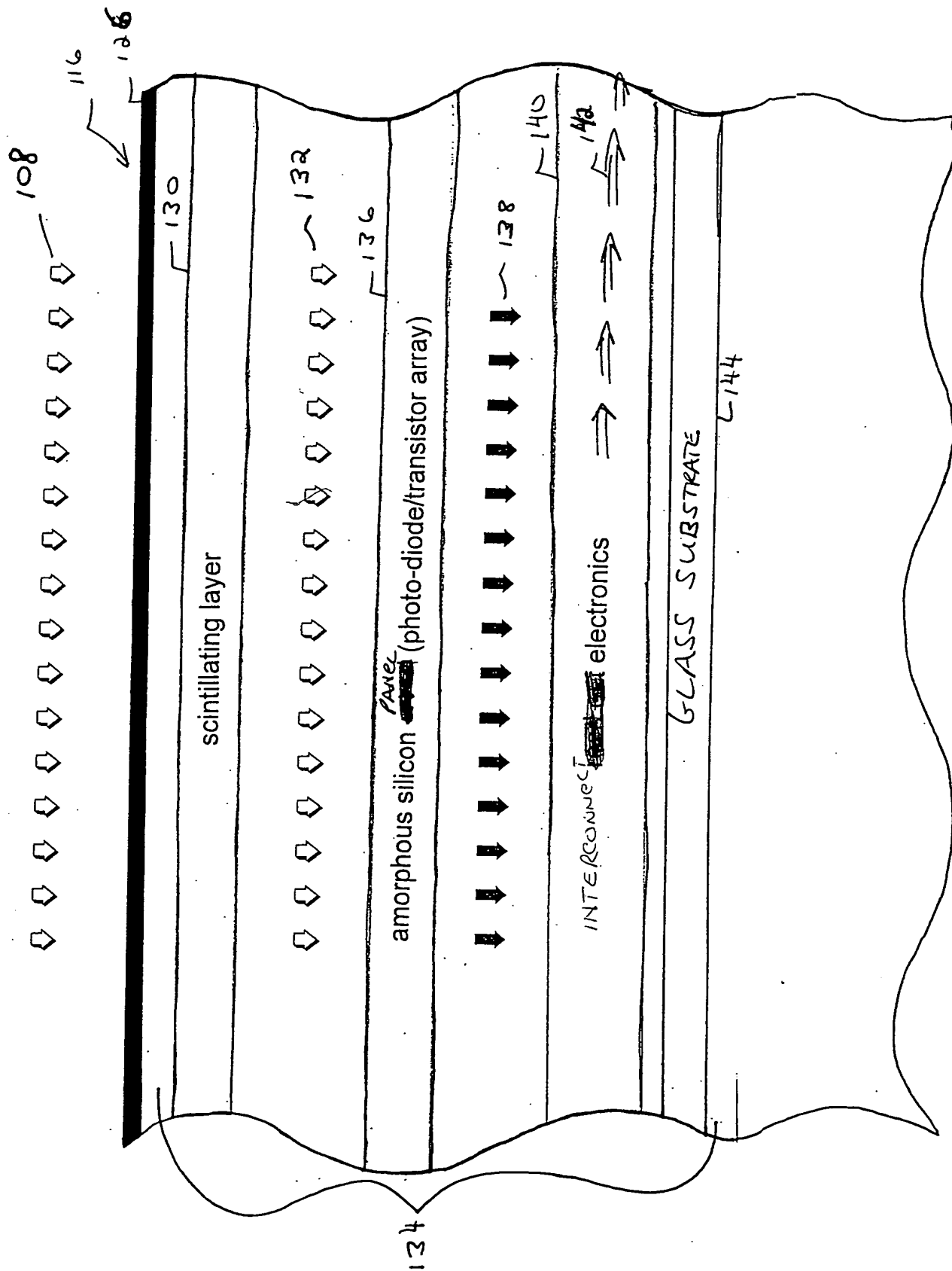
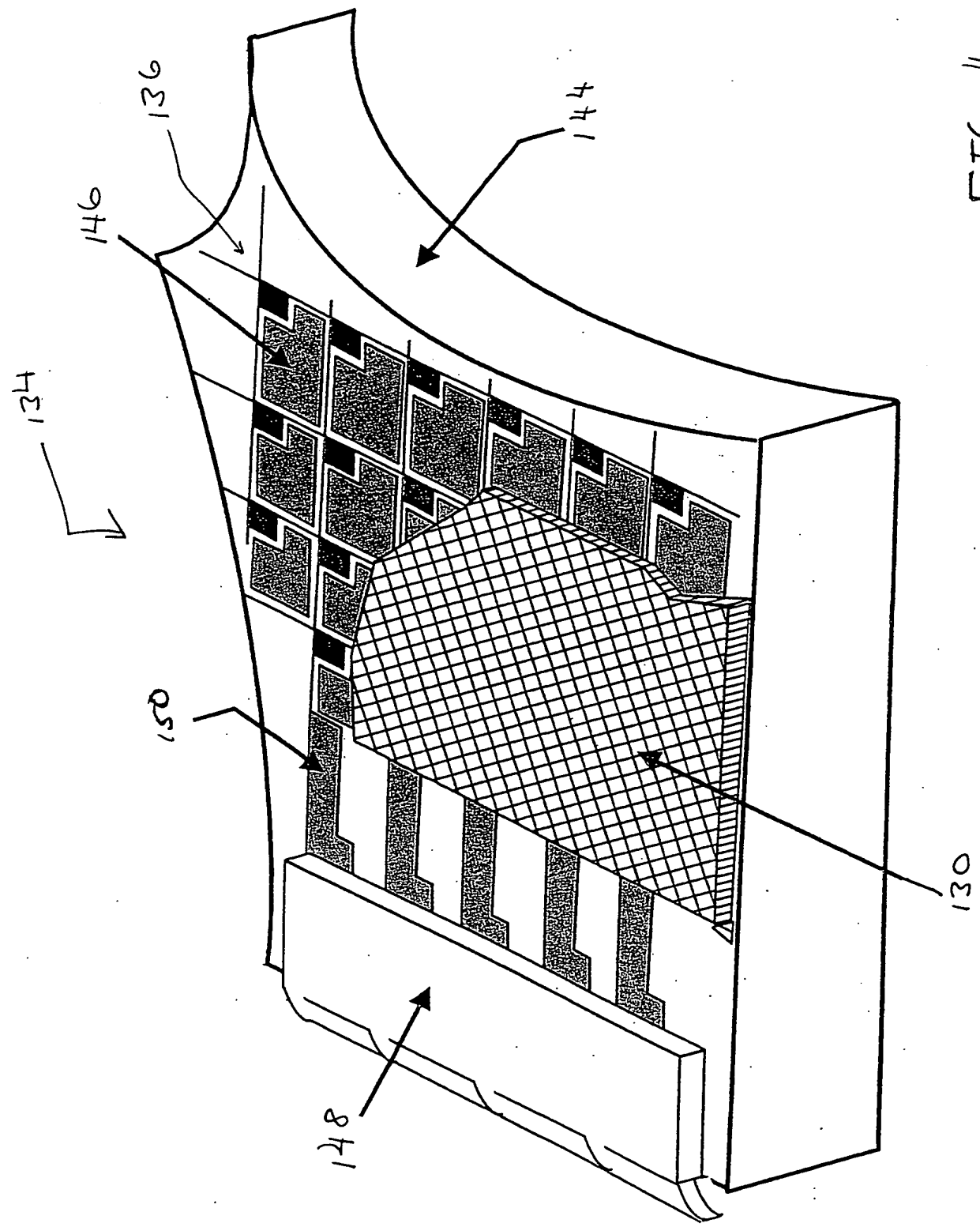


FIG. 3

FIG. 4 is a perspective view of a prior art device.

FIG. 4
(PRIOR ART)



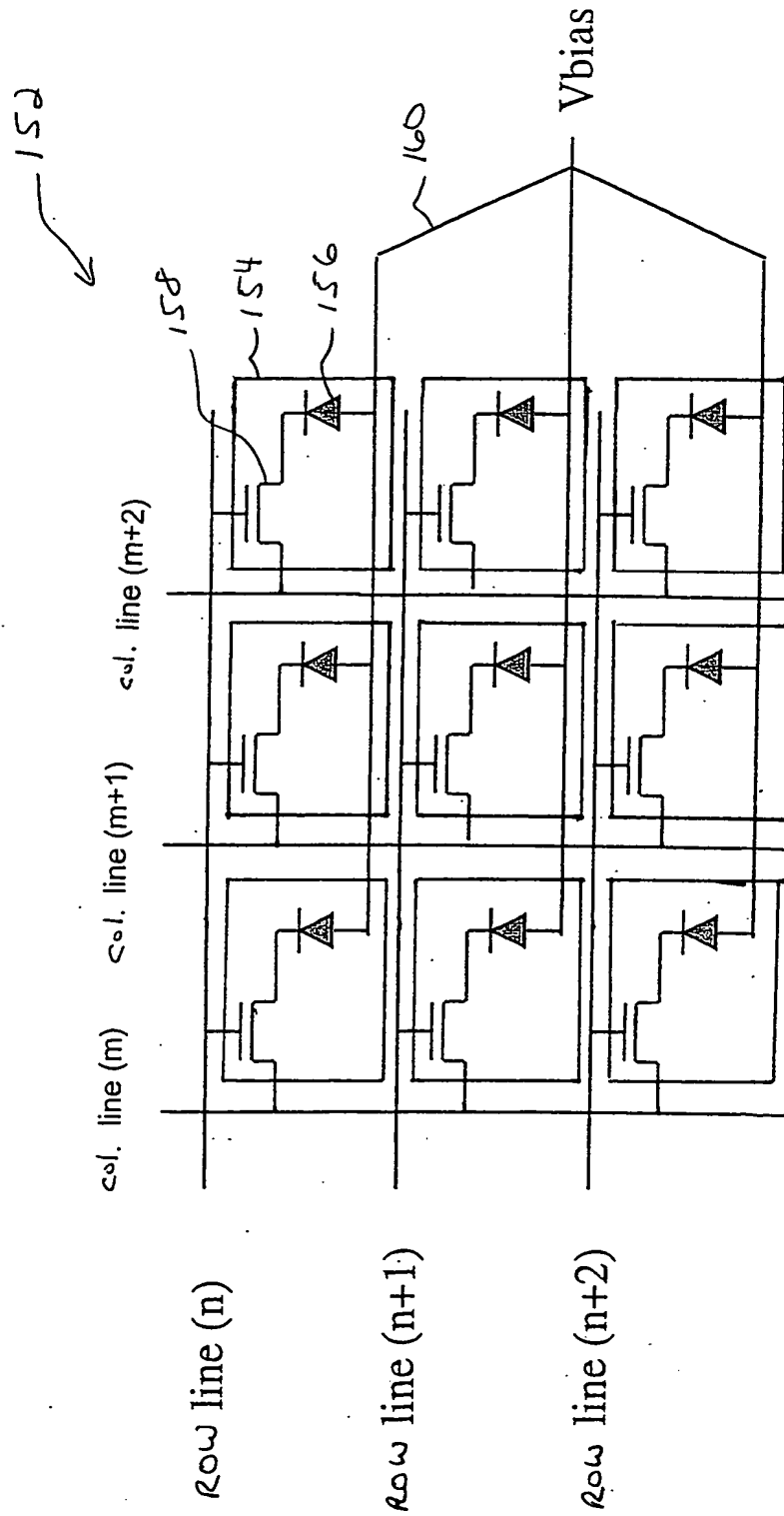


FIG. 5
(PRIOR ART)

FLAT PANEL DETECTOR

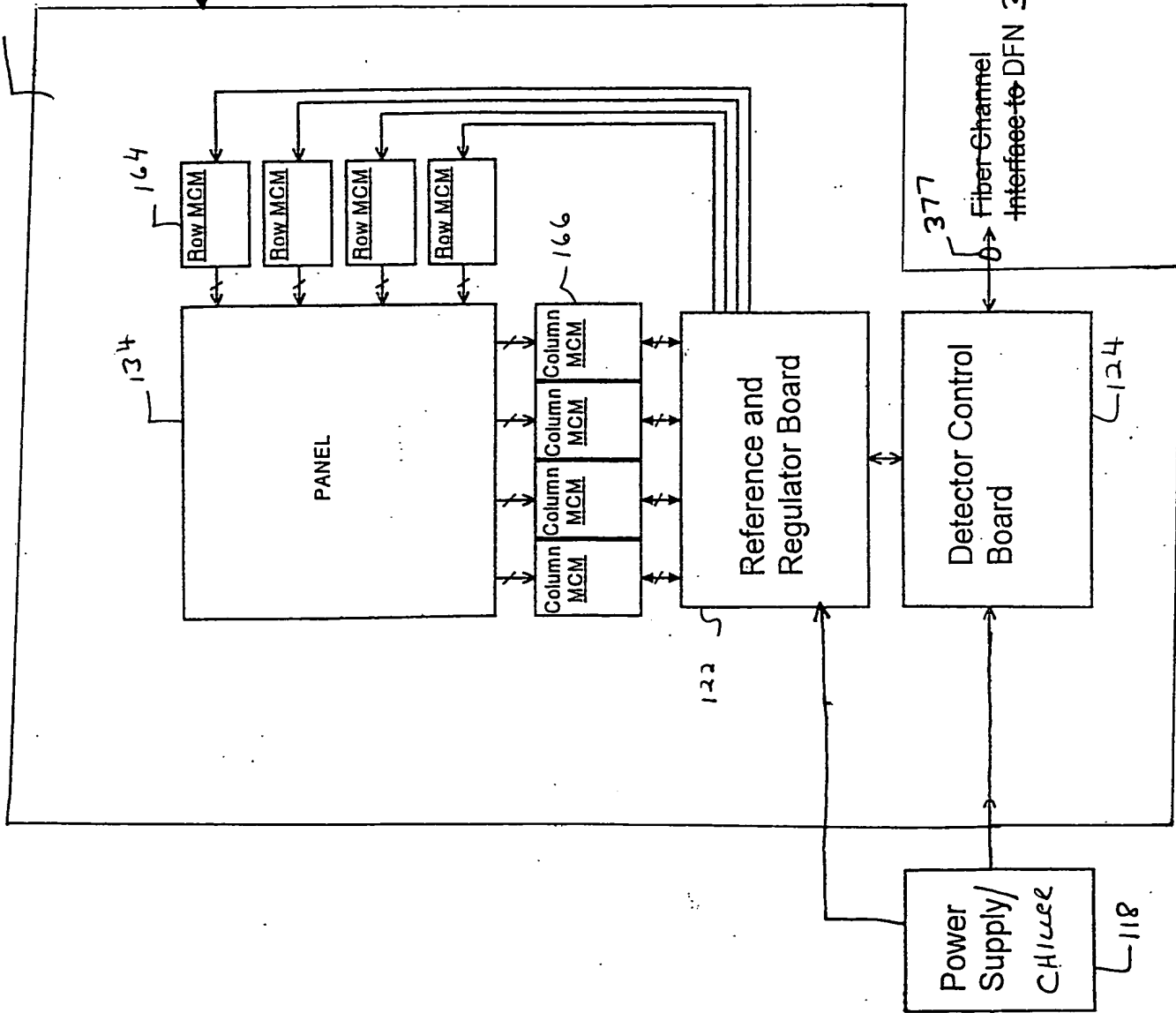


FIG. 6
(Prior Art)

FIG. 7 is a block diagram of a flat panel detector system in accordance with the present invention.

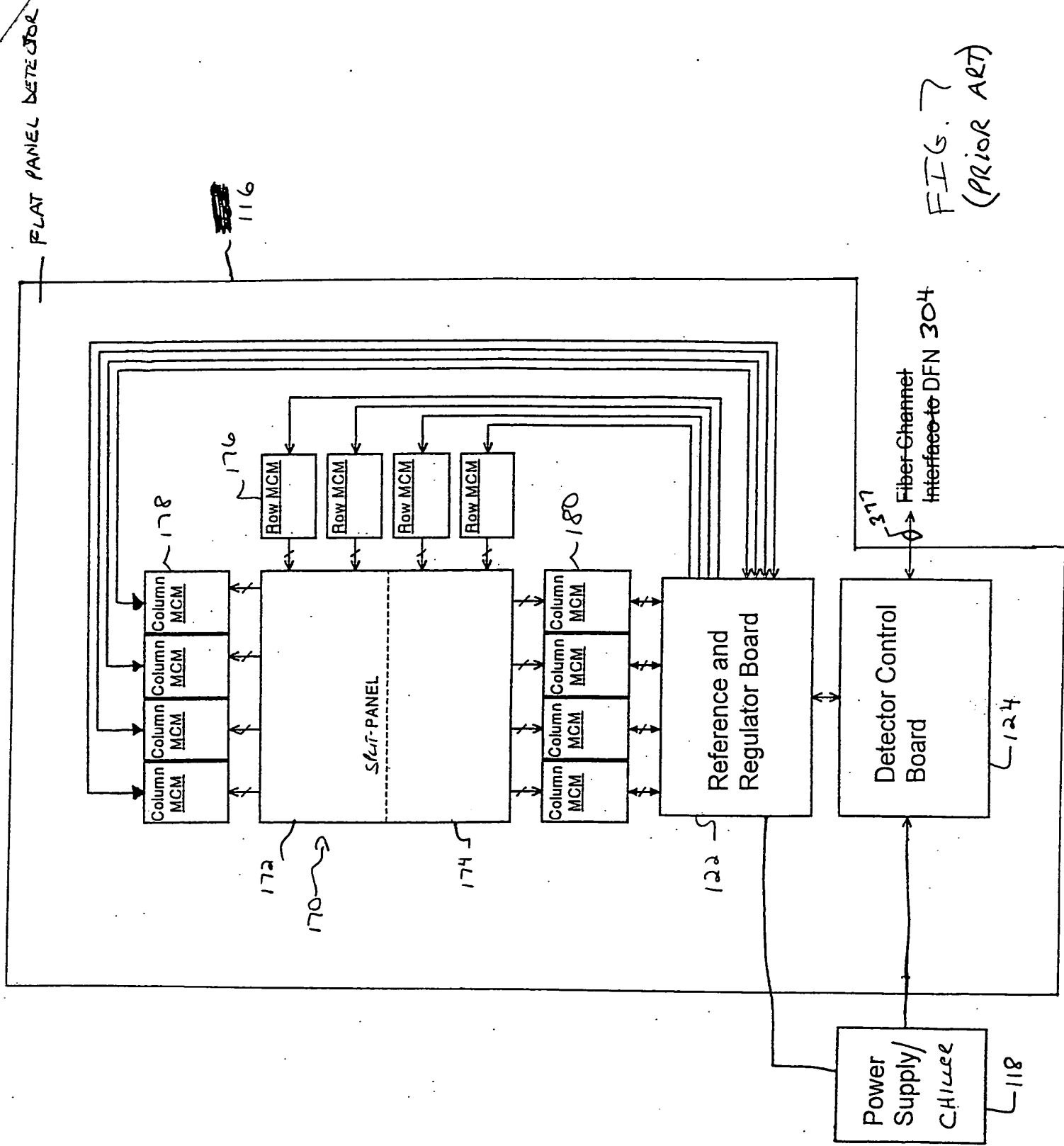
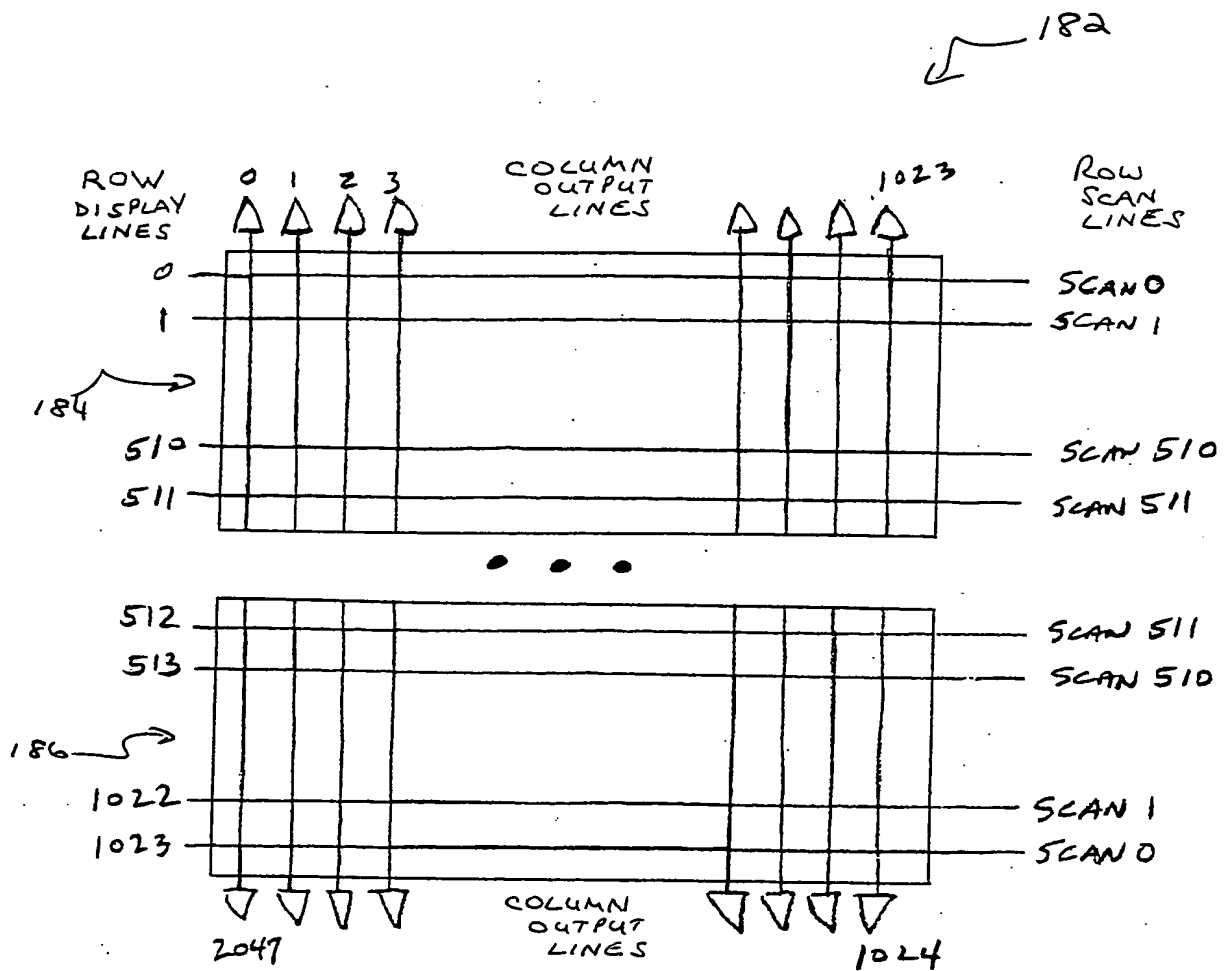


FIG. 7
(PRIOR ART)



CARDIAC/SURGICAL DIGITAL X-RAY PANEL

FIG. 8
(PRIOR ART)

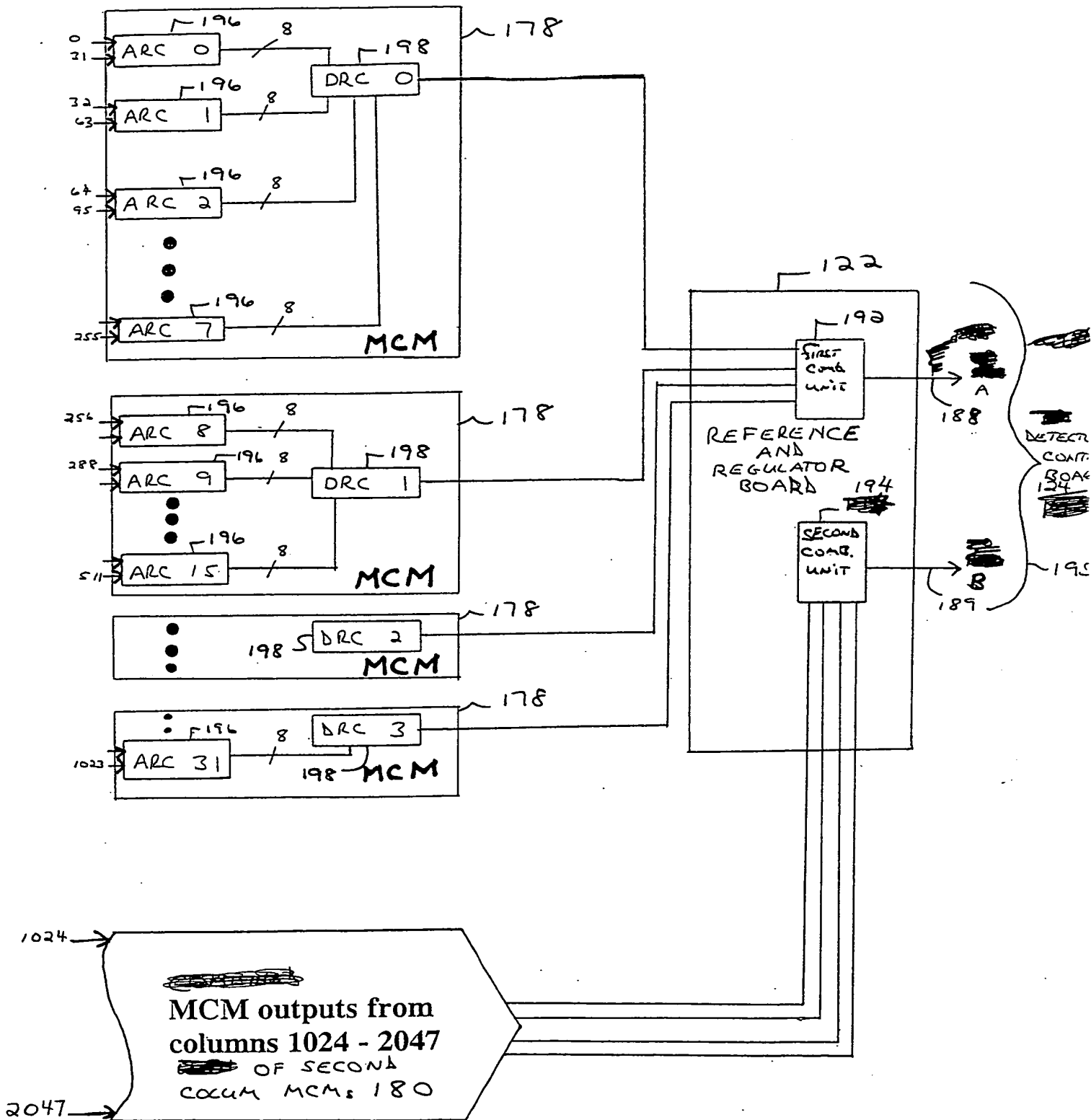


FIG. 9
(PRIOR ART)

FIG. 10 is a block diagram of the detector control board 100, showing the internal components and their interconnections. The board includes an F.O. (Frequency Oscillator) 216, a TRAN. (Transmitter) 214, a REC. (Receiver) 218, an F.O. Sig. Det. (Frequency Oscillator Signal Detector) 222, an ENCODER/DECODER unit 210, a REG. (Register) 220, a CONTROL UNIT 208, two DECODE LUT (Look-Up Table) blocks 204 and 206, and an Osc. (Oscillator) 224. The board is connected to an IMAGE DETECTION BUS 377, an IMAGE COMMUNICATION INTERFACE 124, and a REF. AND REG. BOARD 122. The board also includes a 32-bit data bus 236 and a 12-bit data bus 224.

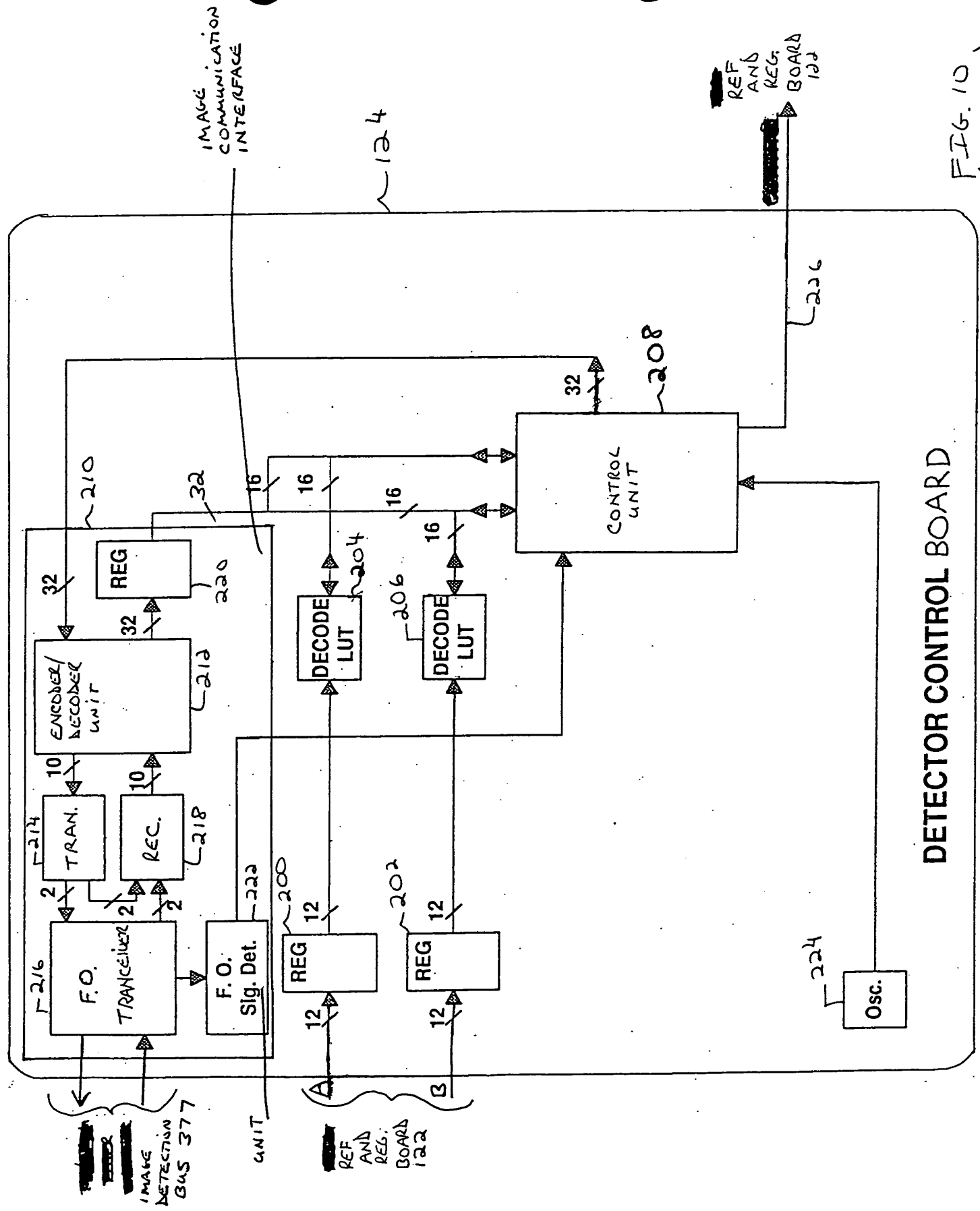
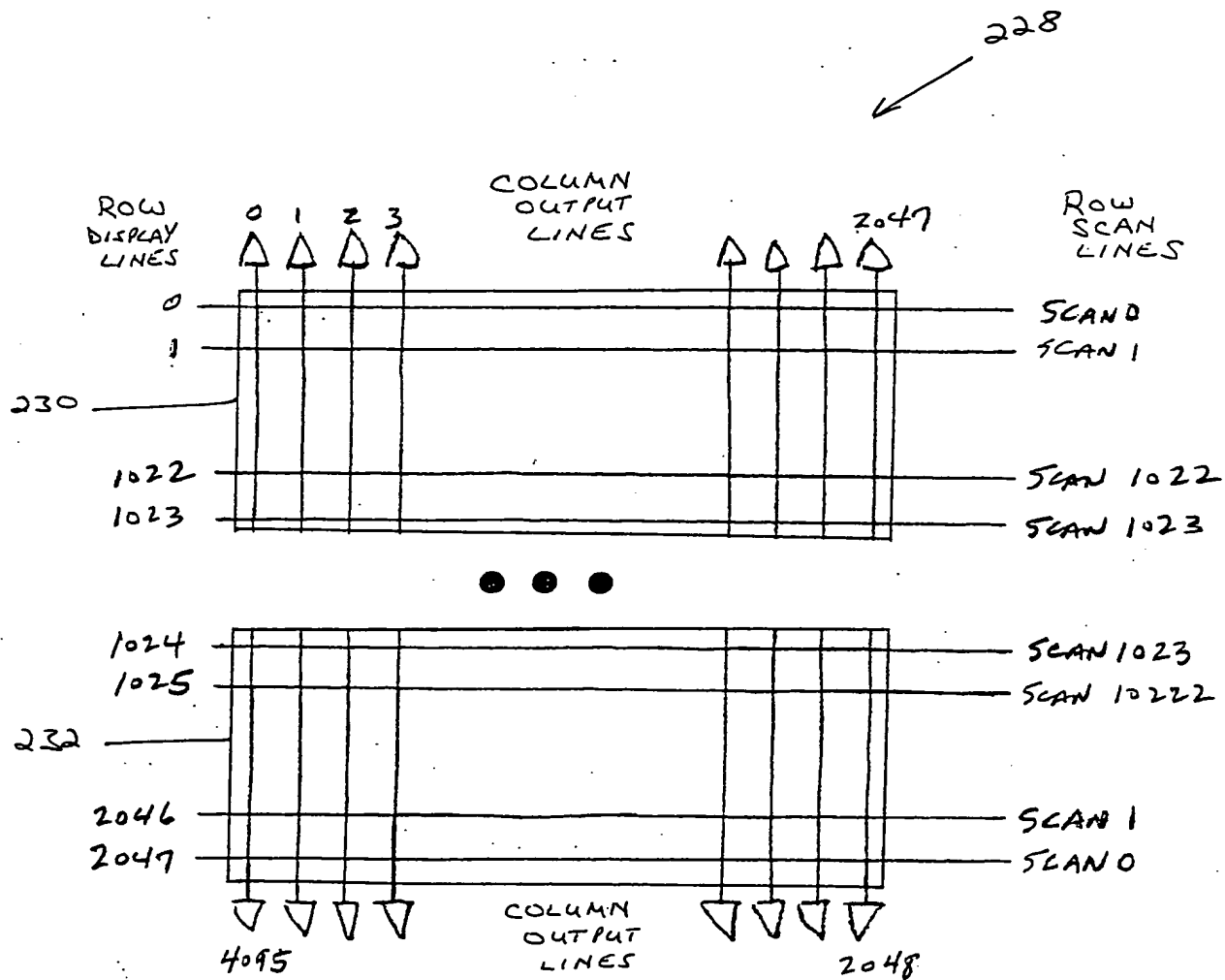


FIG. 10
(PRIOR ART)



RADIOGRAPHY DIGITAL X-RAY PANEL

FIG. 11
(PRIOR ART)

FIG. 12 is a block diagram of a flat panel detector system. The system includes a flat panel detector 116, a reference and regulator board 122, a detector control board 124, and a power supply/charger 118. The flat panel detector 116 is connected to the reference and regulator board 122 via a fiber channel interface 304. The reference and regulator board 122 is connected to the detector control board 124. The power supply/charger 118 provides power to the detector control board 124.

FLAT PANEL
DETECTOR

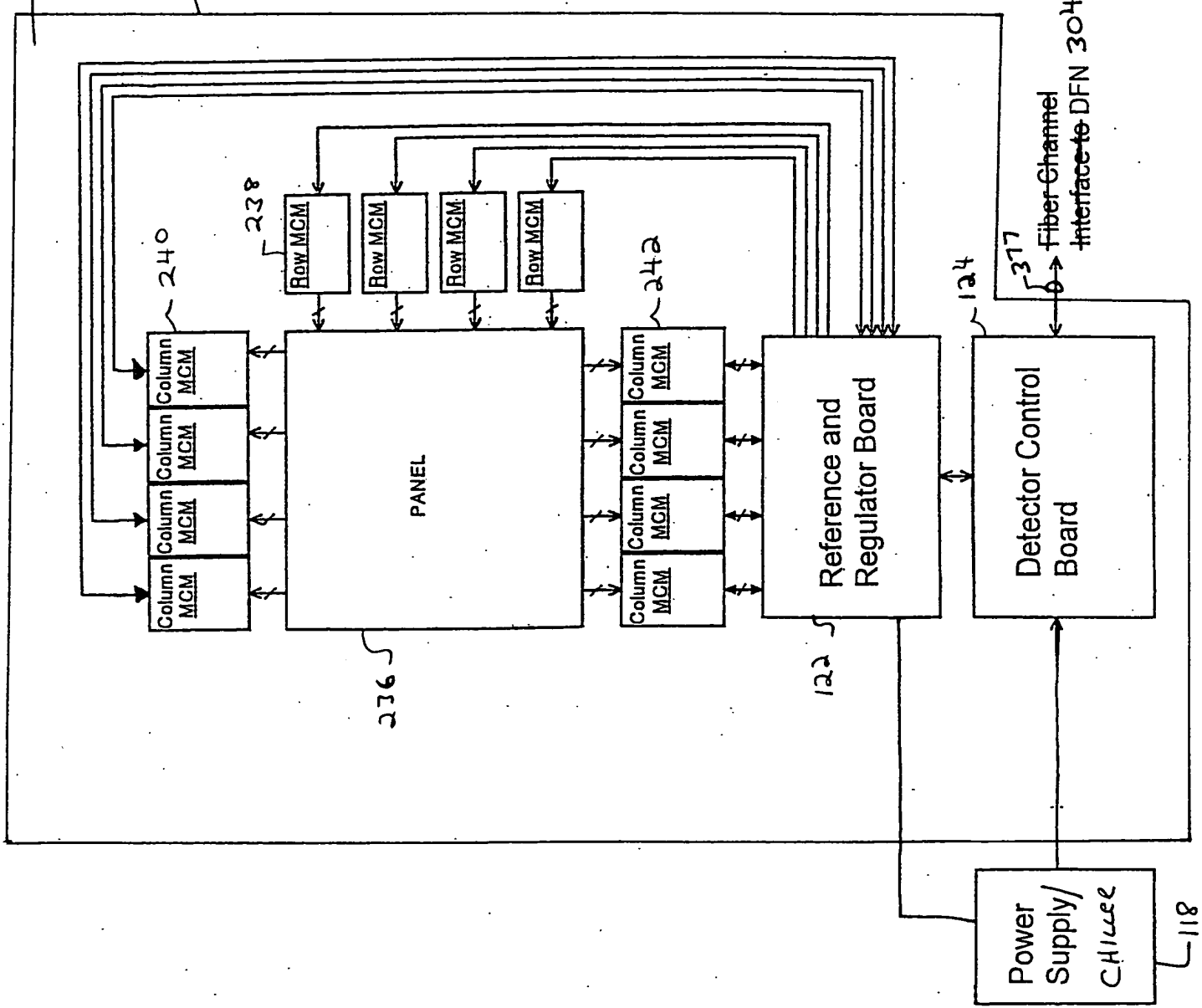
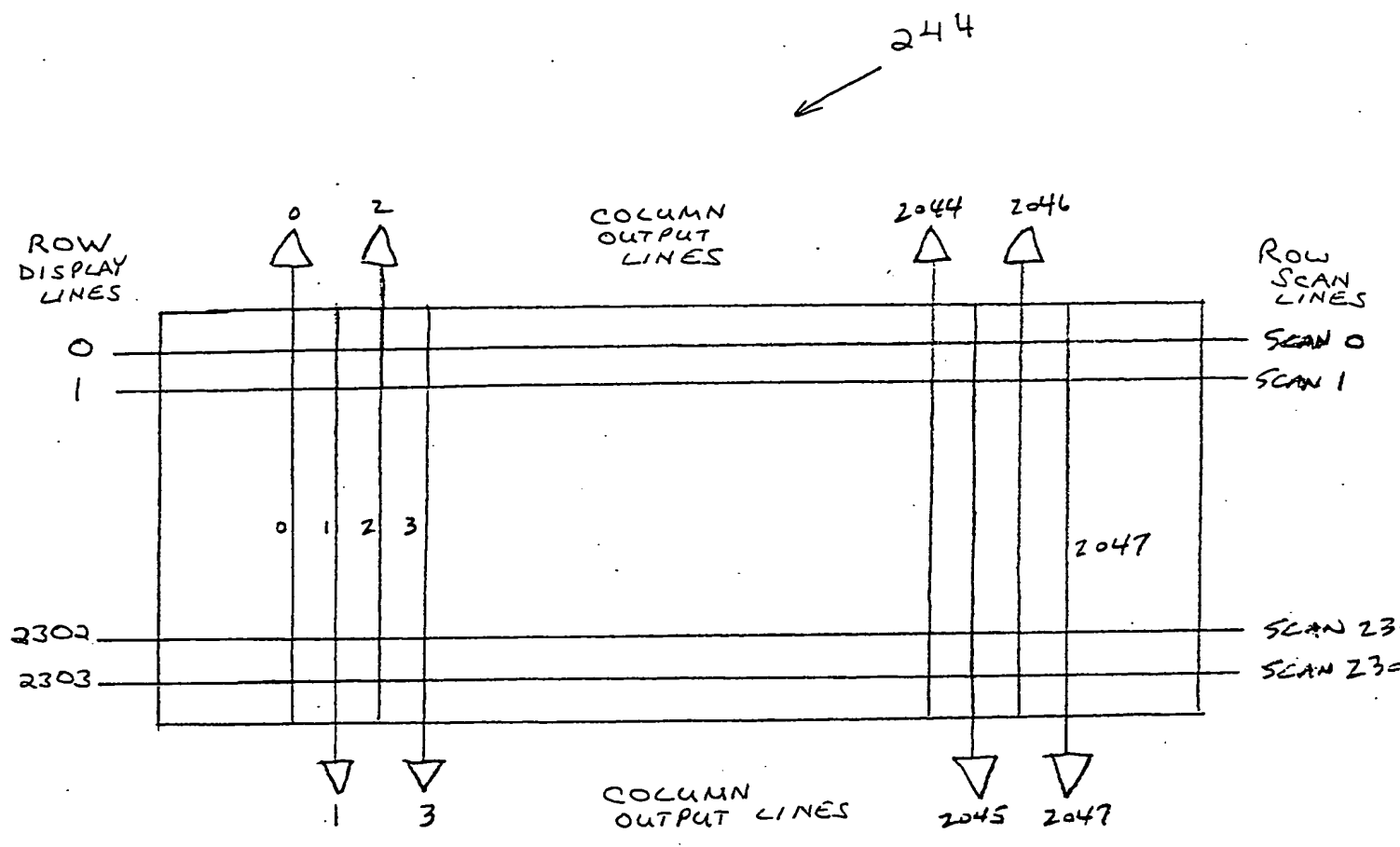


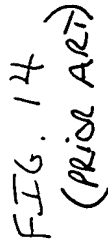
FIG. 12
(Prior Art)

FIG. 13 (PRIOR ART) is a schematic diagram of a mamography digital x-ray panel. The diagram shows a grid of 24 rows and 24 columns. The rows are labeled 0 to 23 on the left and 240 to 263 on the right. The columns are labeled 0 to 23 on the top and 240 to 263 on the bottom. The grid is divided into four quadrants by a horizontal line between rows 11 and 12, and a vertical line between columns 11 and 12. The top-left quadrant is labeled '0' and the bottom-right quadrant is labeled '2047'. The top-right quadrant is labeled '2044' and the bottom-left quadrant is labeled '2045'.



MAMOGRAPHY DIGITAL X-RAY PANEL

FIG. 13
(PRIOR ART)

[illegible]

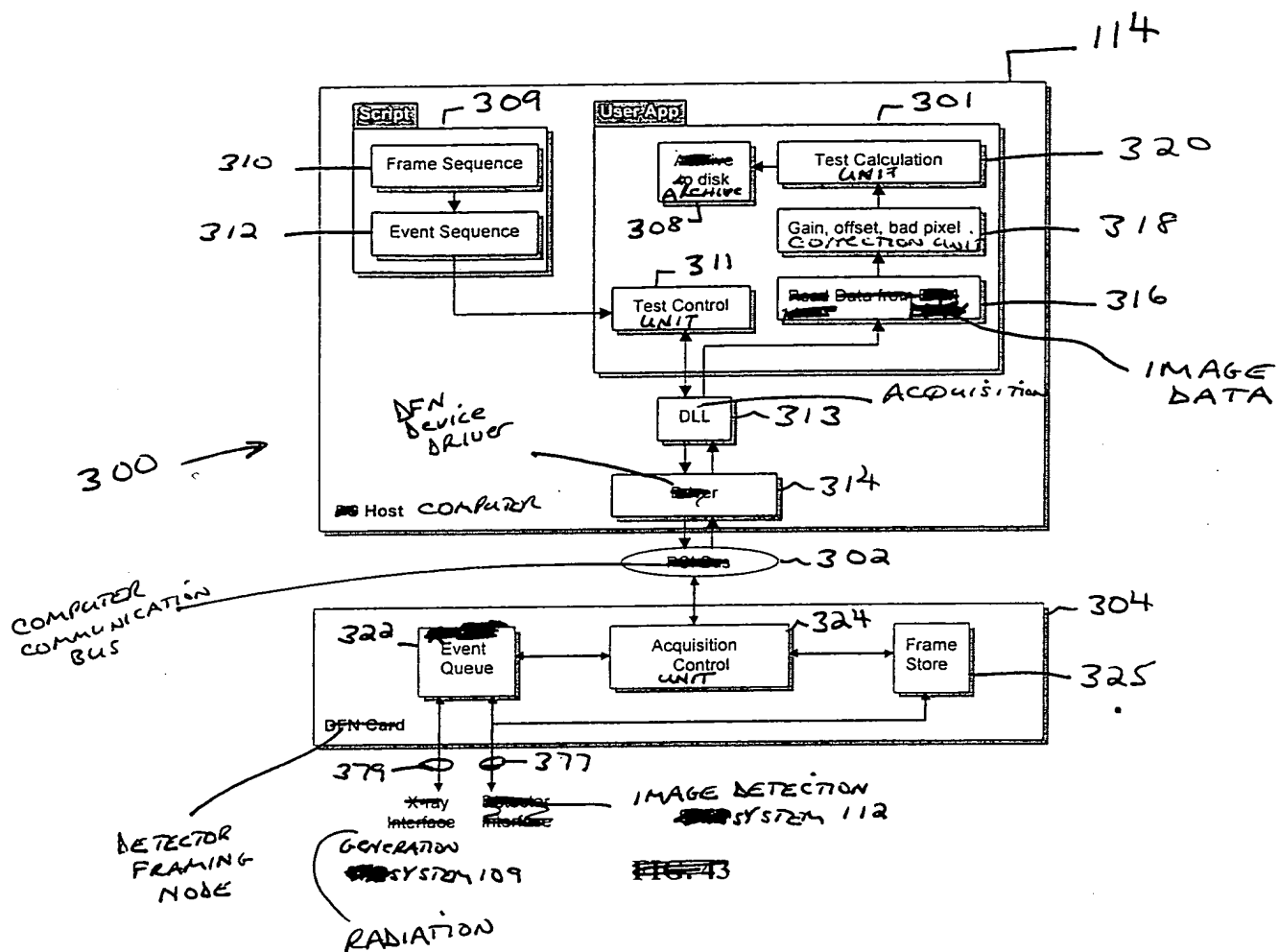


FIG. 15

w:\1346\todd-dfn cases\drawings\dwg-01 (page 3)

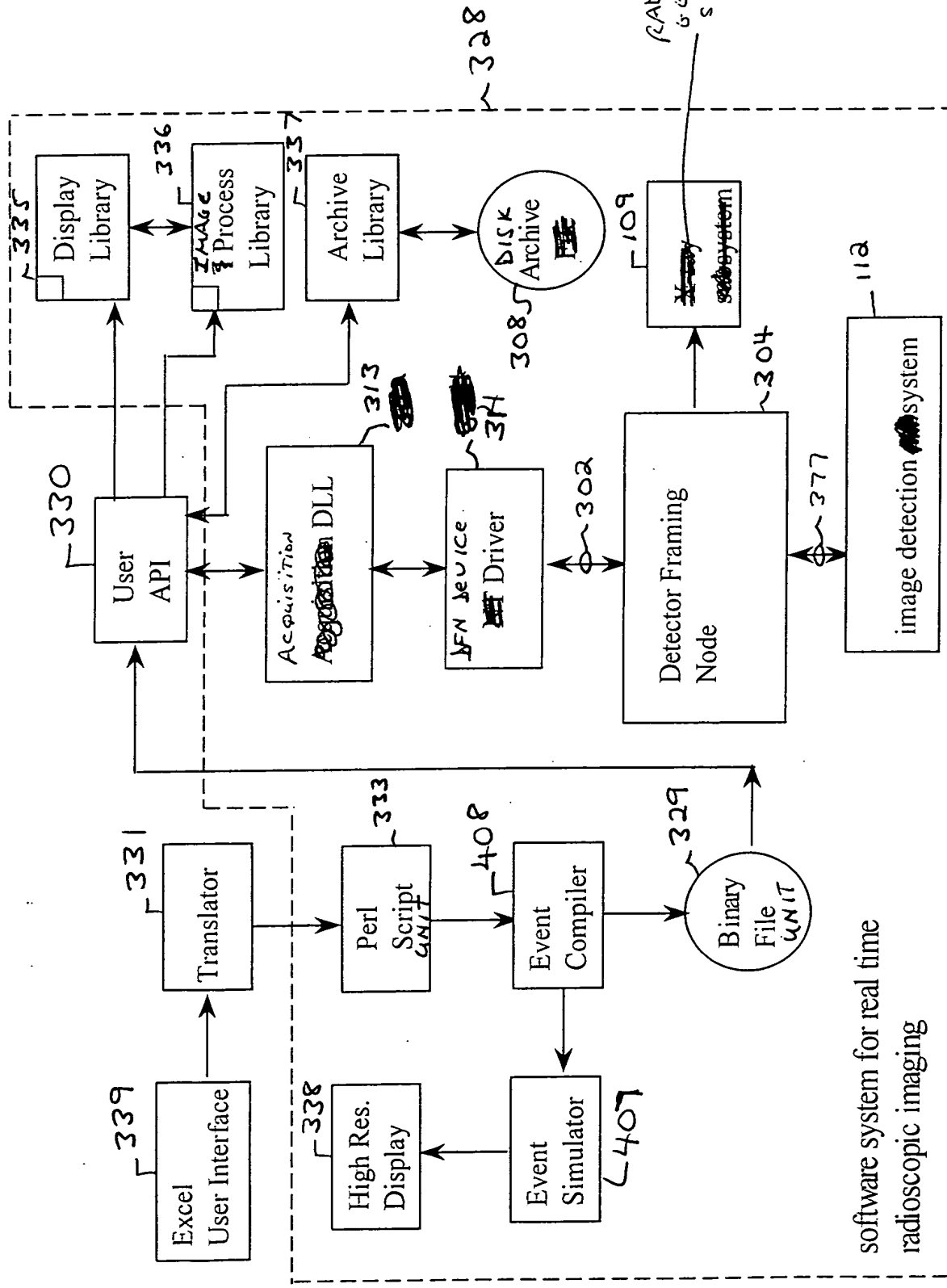


FIG. 16

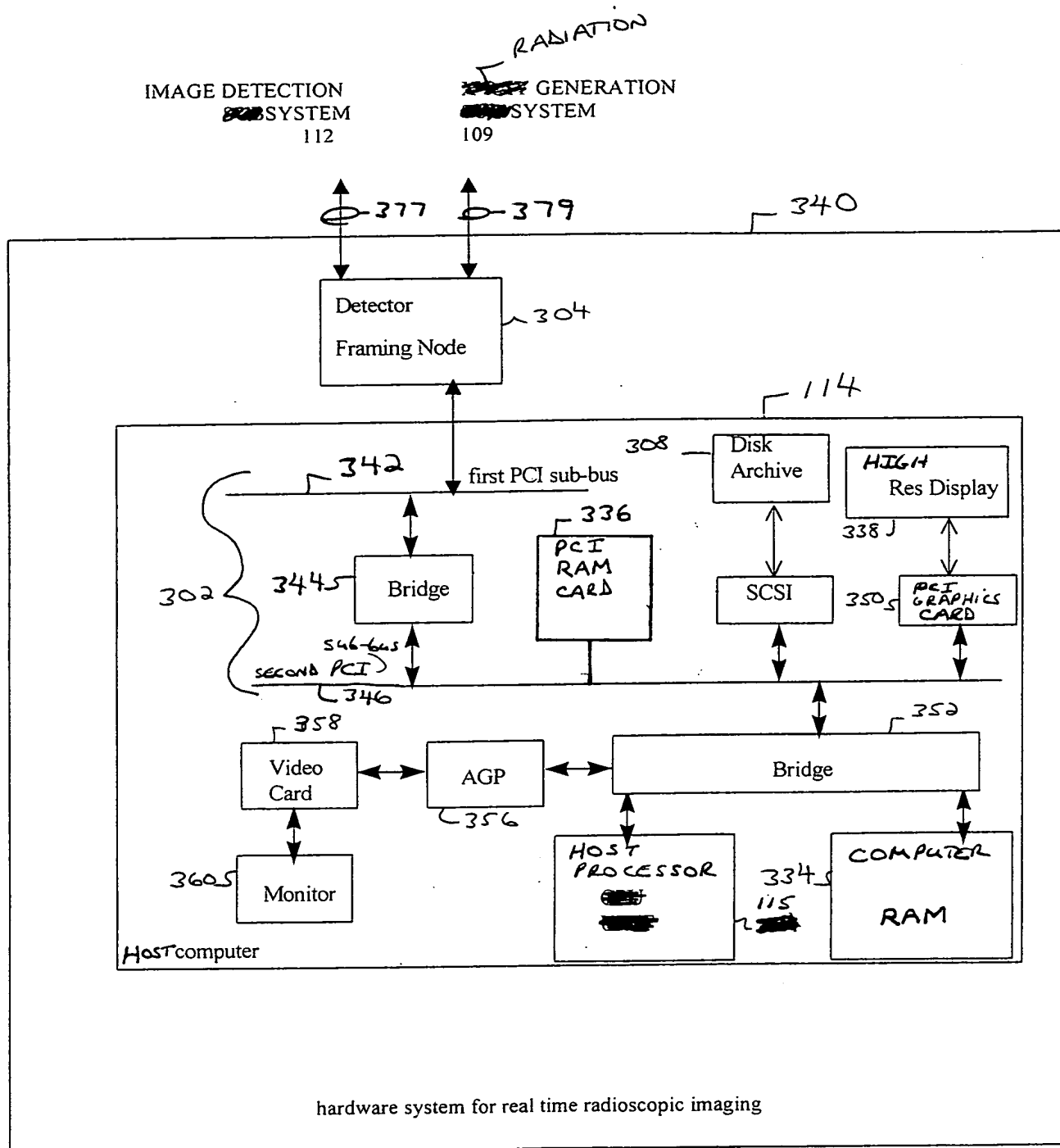
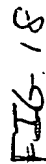


FIG. 17

$$\left. \begin{array}{l} 2 \\ 1 \end{array} \right\}$$


W:\1346\odd-DFN cases\drawings\estimated image processing.doc

Panel Setup	Real Time	(fm/sec)	length	Latency	memory	offset	gbr
Single Frame	Post Process	30	unlimited	< 5 frames	host	none	
Single Frame	Post Process	-	-	Delay ~.1 sec	"	y	
Single Frame	Post Process	-	-	Delay ~.2 sec	"	y	y
Real Time	Real Time	R	Unlimited	< 5 frames	host	none	
Real Time	Real Time	R - X	Unlimited	< 5 frames	"	y	
Real Time	Real Time	R - Y	Unlimited	< 5 frames	"	y	y

FIG-19

Modality	image size	Frames Stored
Cardiac	1024 x 1024	host memory
Rad	2048 x 2048	200
Mammo	2304 x 2048	50
		44

FIG. 20

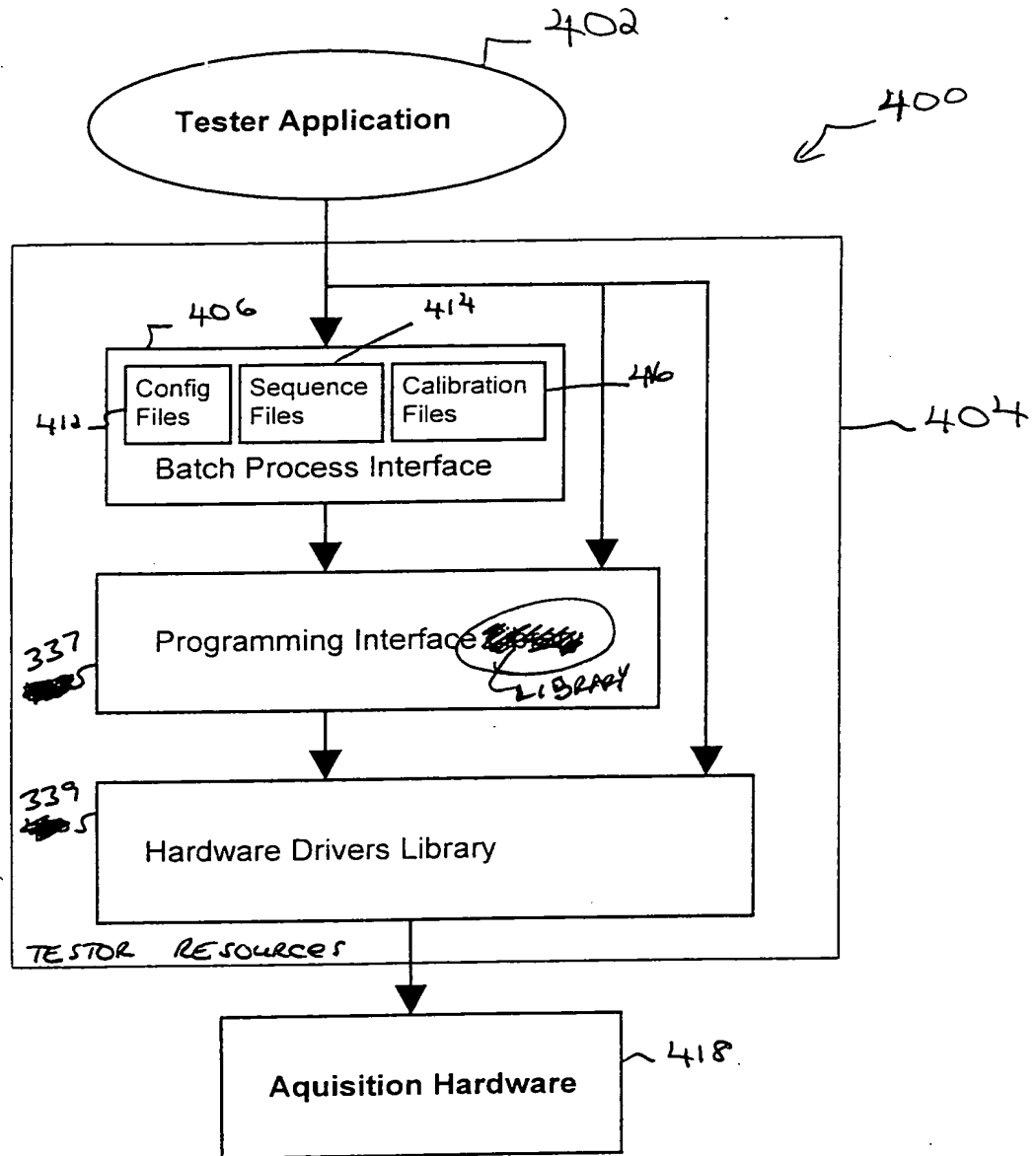


FIG. 21

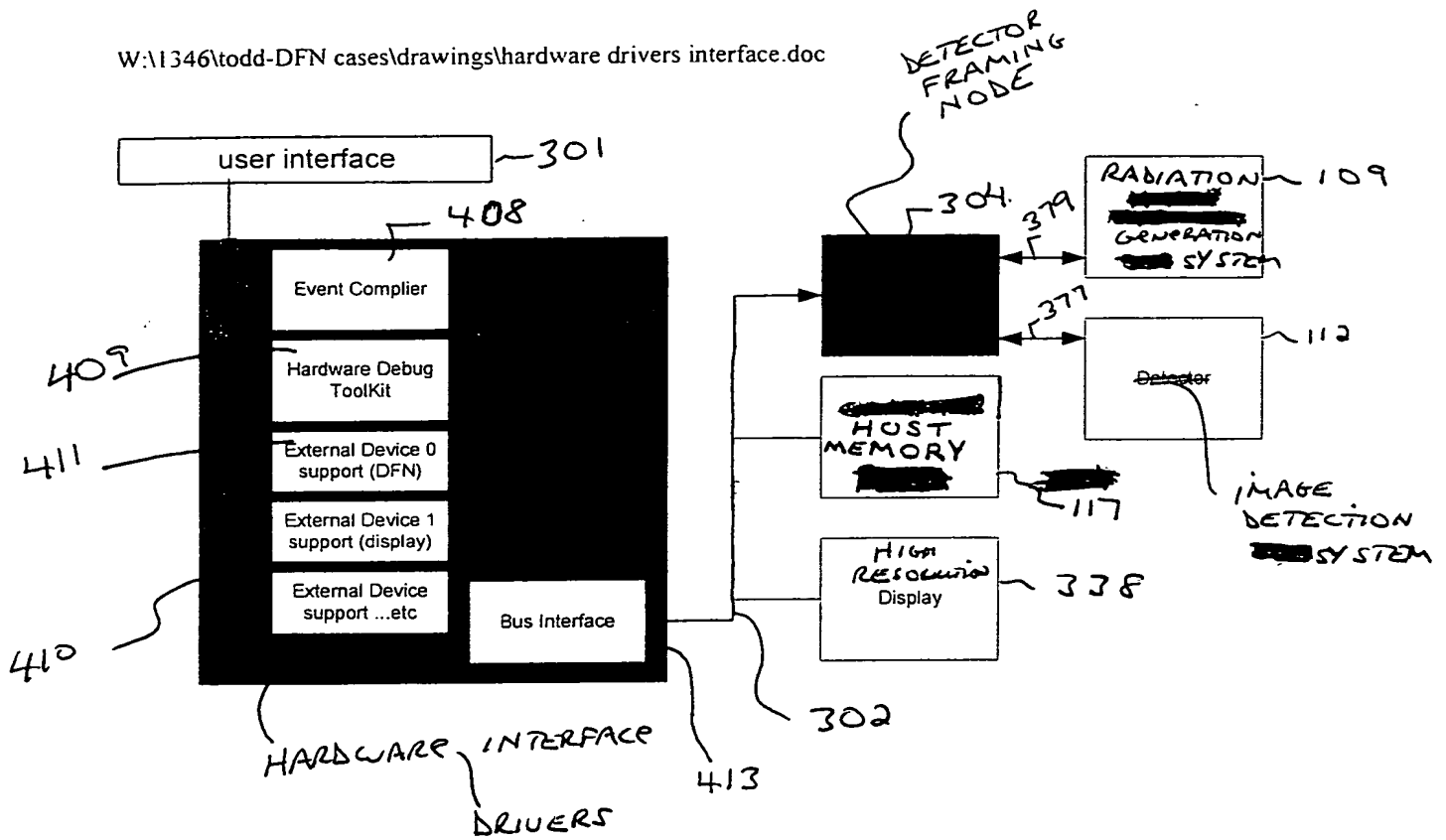


FIG. 22

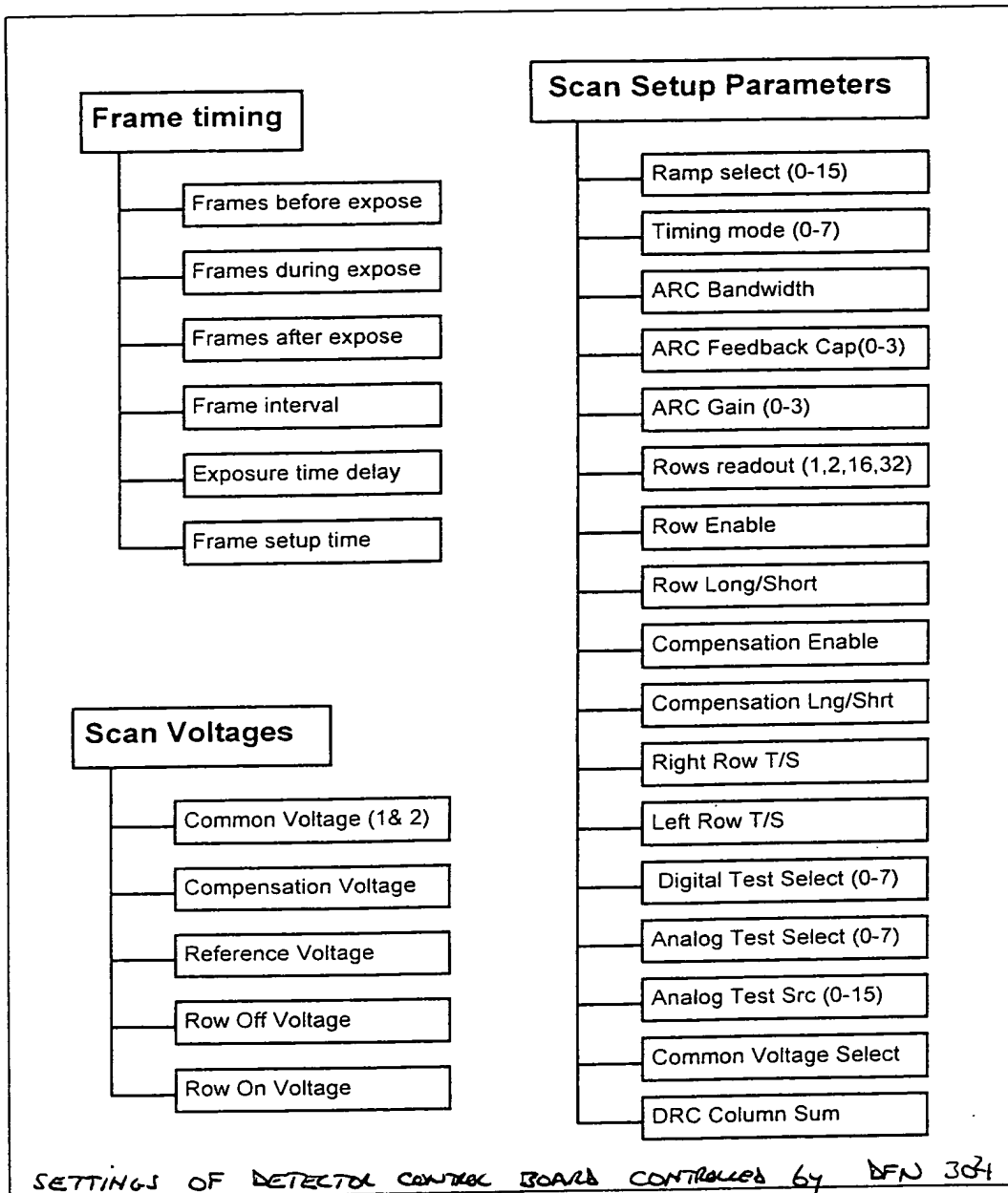


FIG. 23

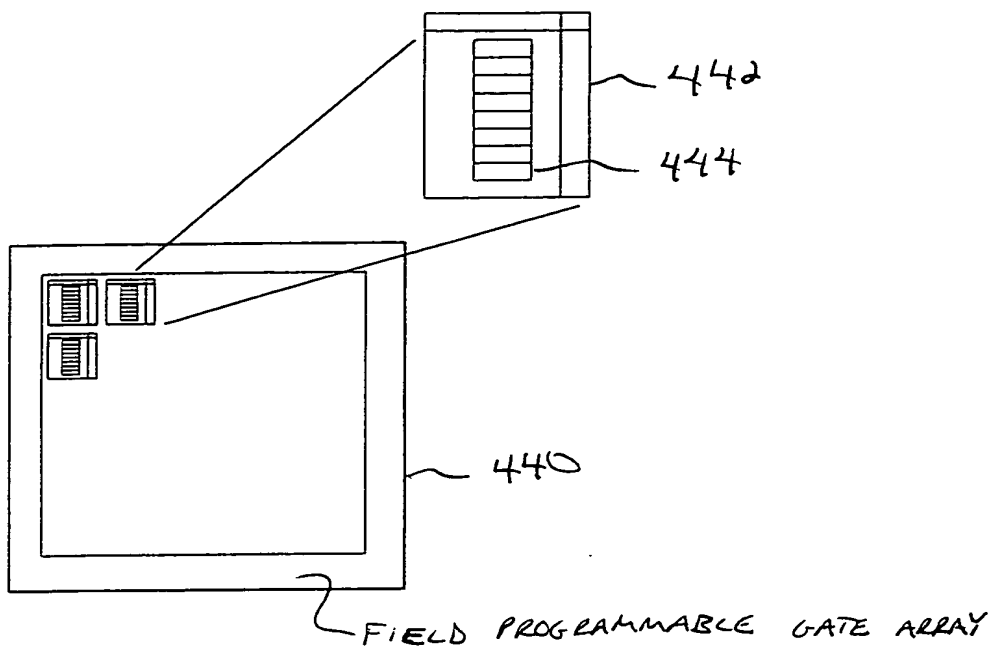


FIG. ~~23~~ 24

~~DETECTOR~~
CONTROL BOARD 124

EVENT
PROCESSOR

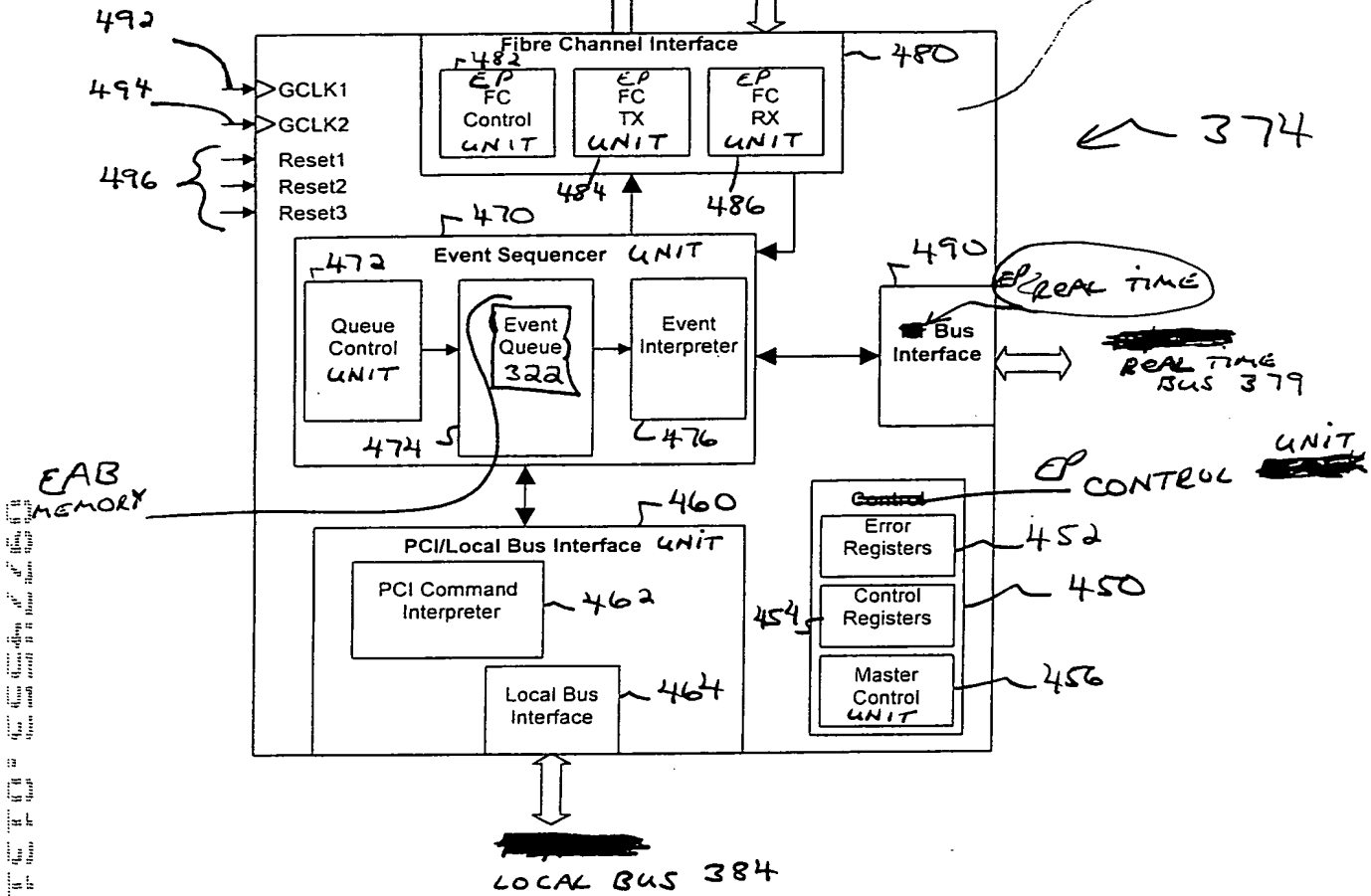


FIG. 25

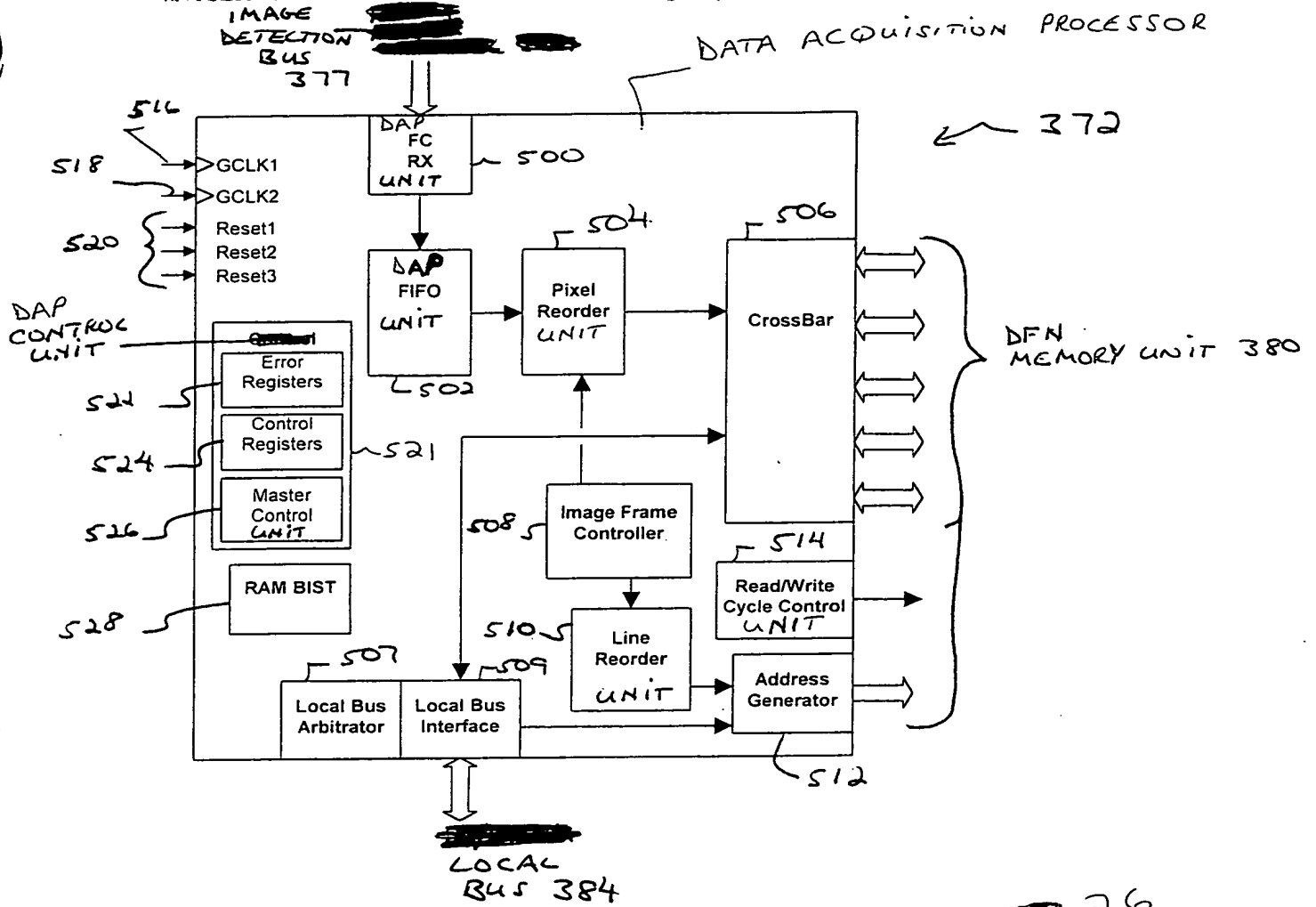


FIG. 26

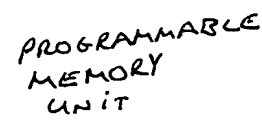


FIG. ~~26~~ 27

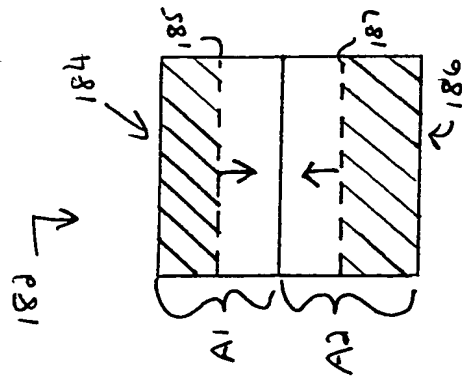


FIG. 28A
28

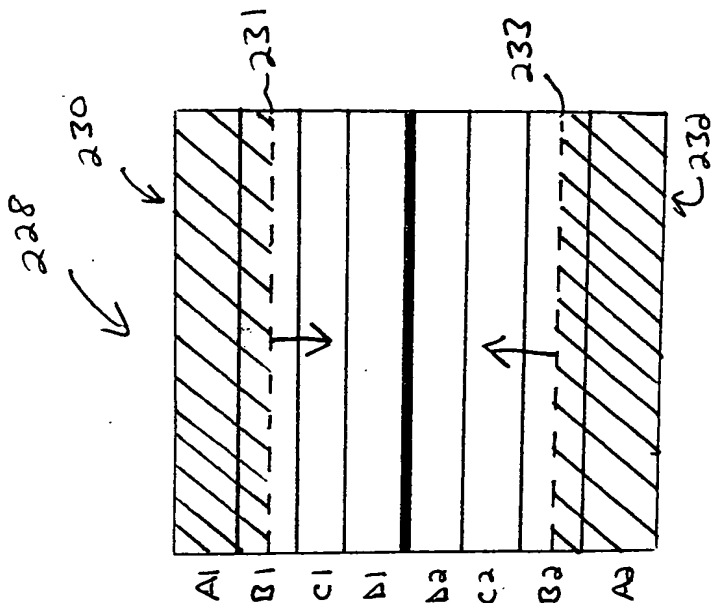


FIG. 28B
29

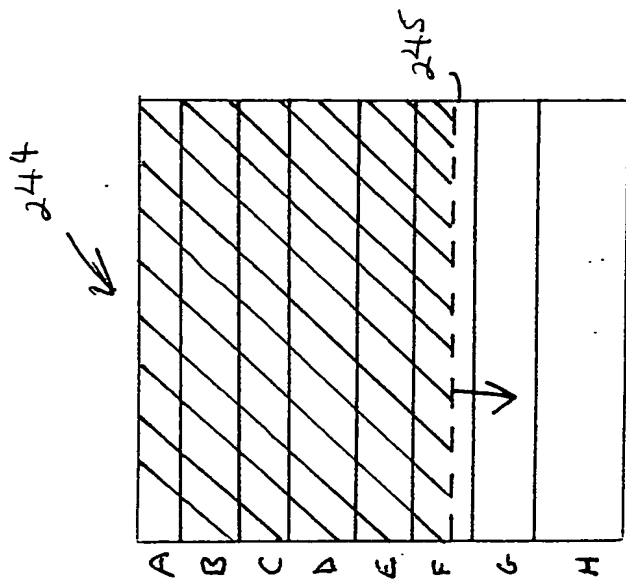


FIG. 28C
30

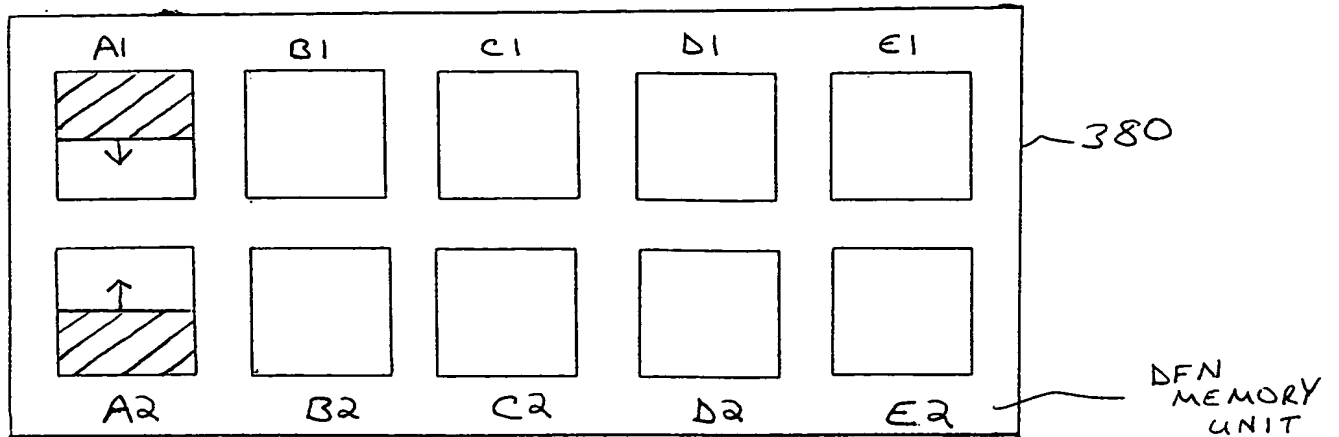


FIG. ~~30A~~ 31

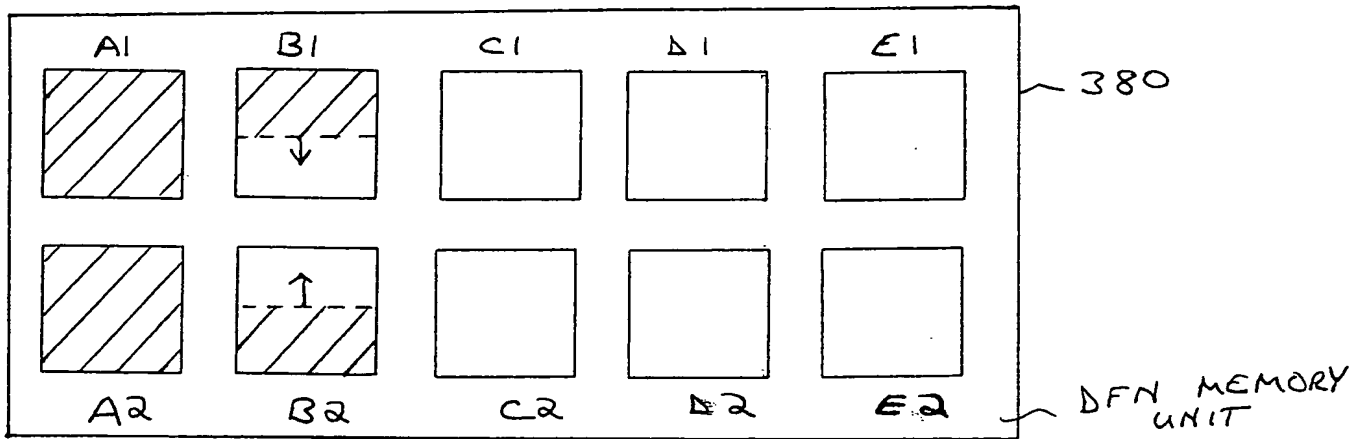


FIG. ~~30B~~ 32

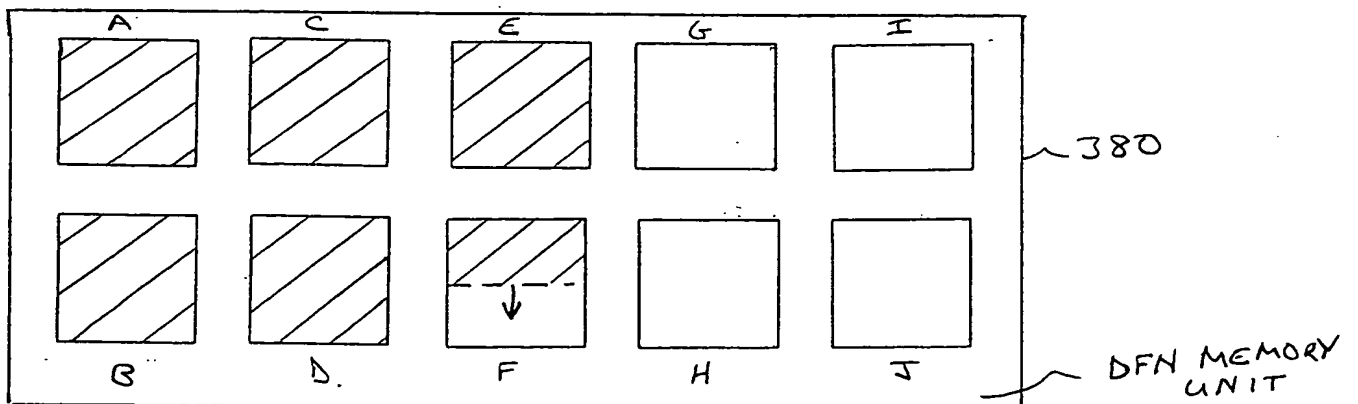


FIG. ~~30C~~ 33

334

A1
A2

FIG. ~~34~~ 34

334

A1
B1
C1
A1
A2
C2
B2
A2

FIG. ~~35~~ 35

334

A
B
C
A
E
F
G
H

FIG. ~~36~~ 36

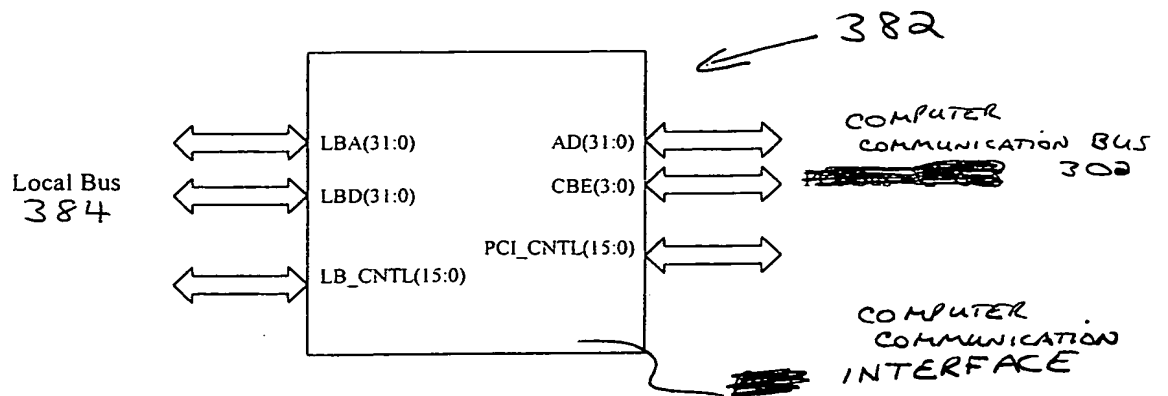


FIG. 37

TOP SECRET

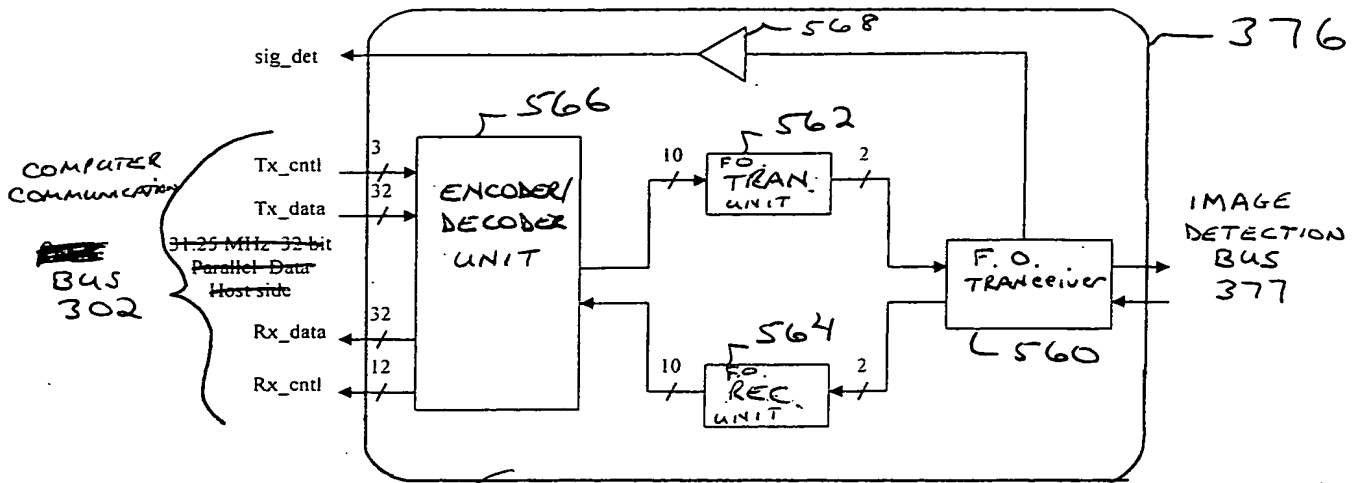


IMAGE DETECTION INTERFACE

FIG. 38

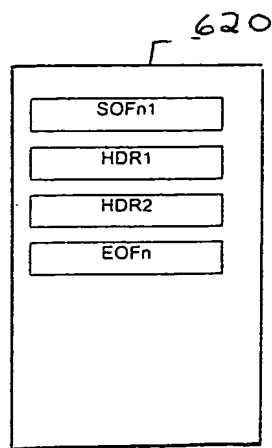


FIG. ~~38~~
39

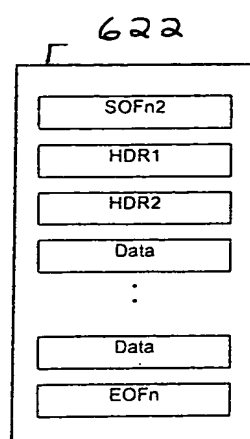


FIG. ~~39~~
40

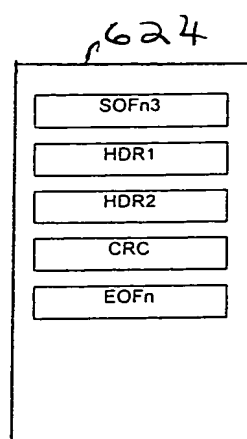


FIG. ~~40~~
41

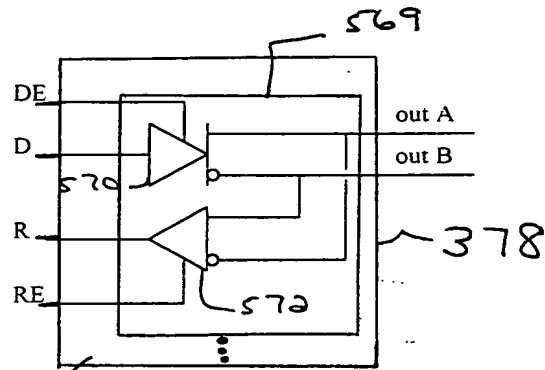
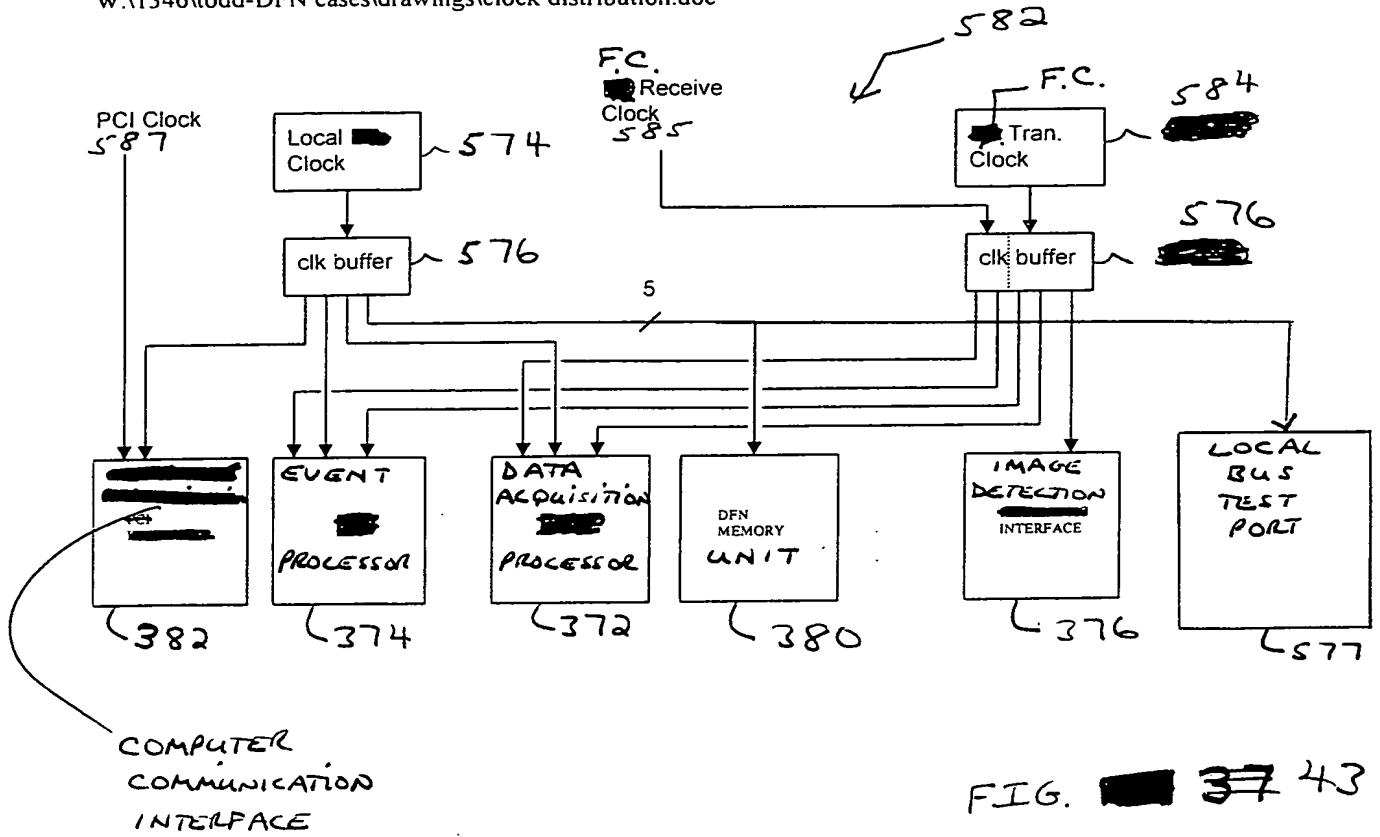


FIG. ~~41~~ ~~42~~ 42

REAL TIME
BUS INTERFACE

TOP SECRET



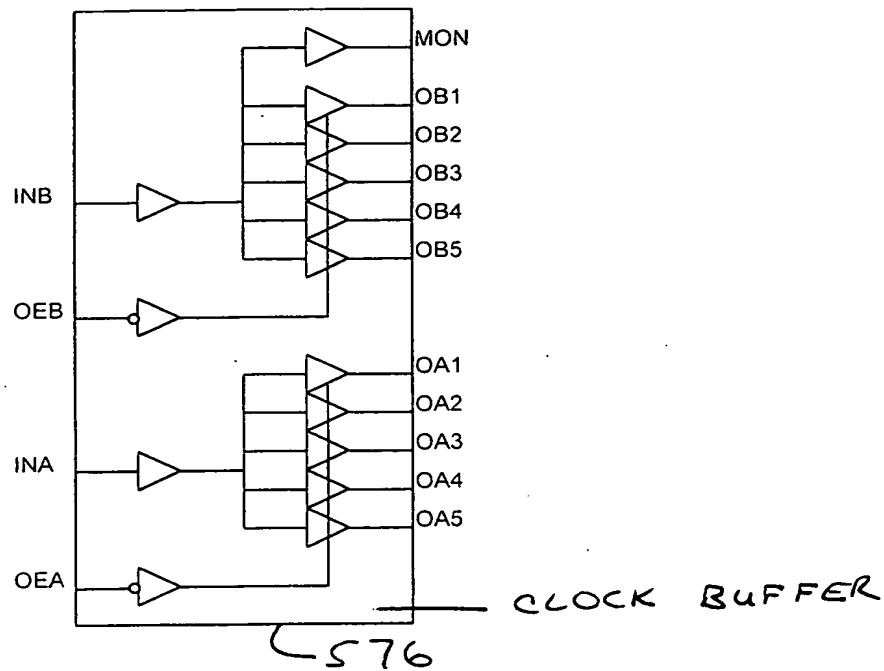


FIG. ~~33~~ 44

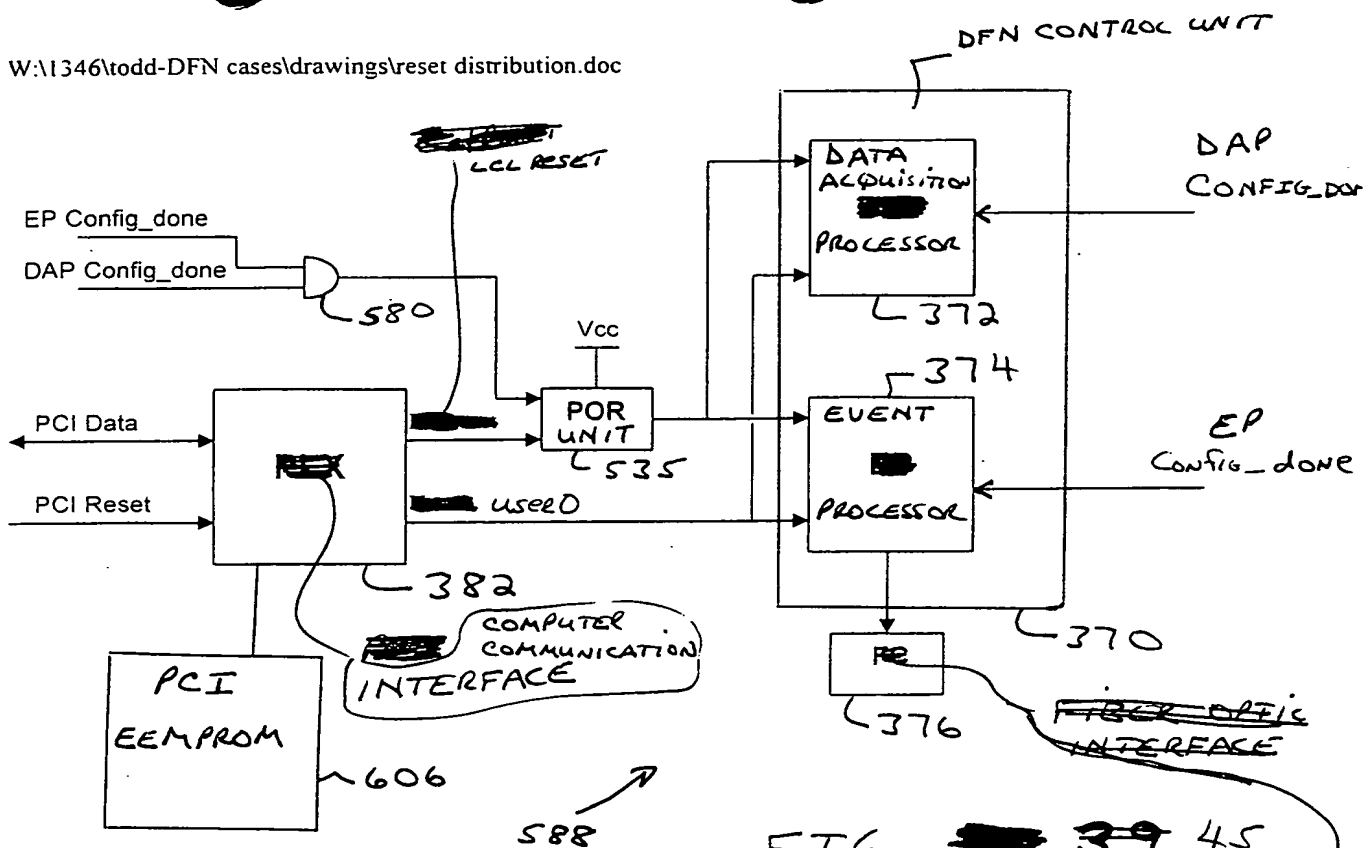


FIG. 39 45

IMAGE DETECTION INTERFACE
COMPUTER COMMUNICATION INTERFACE

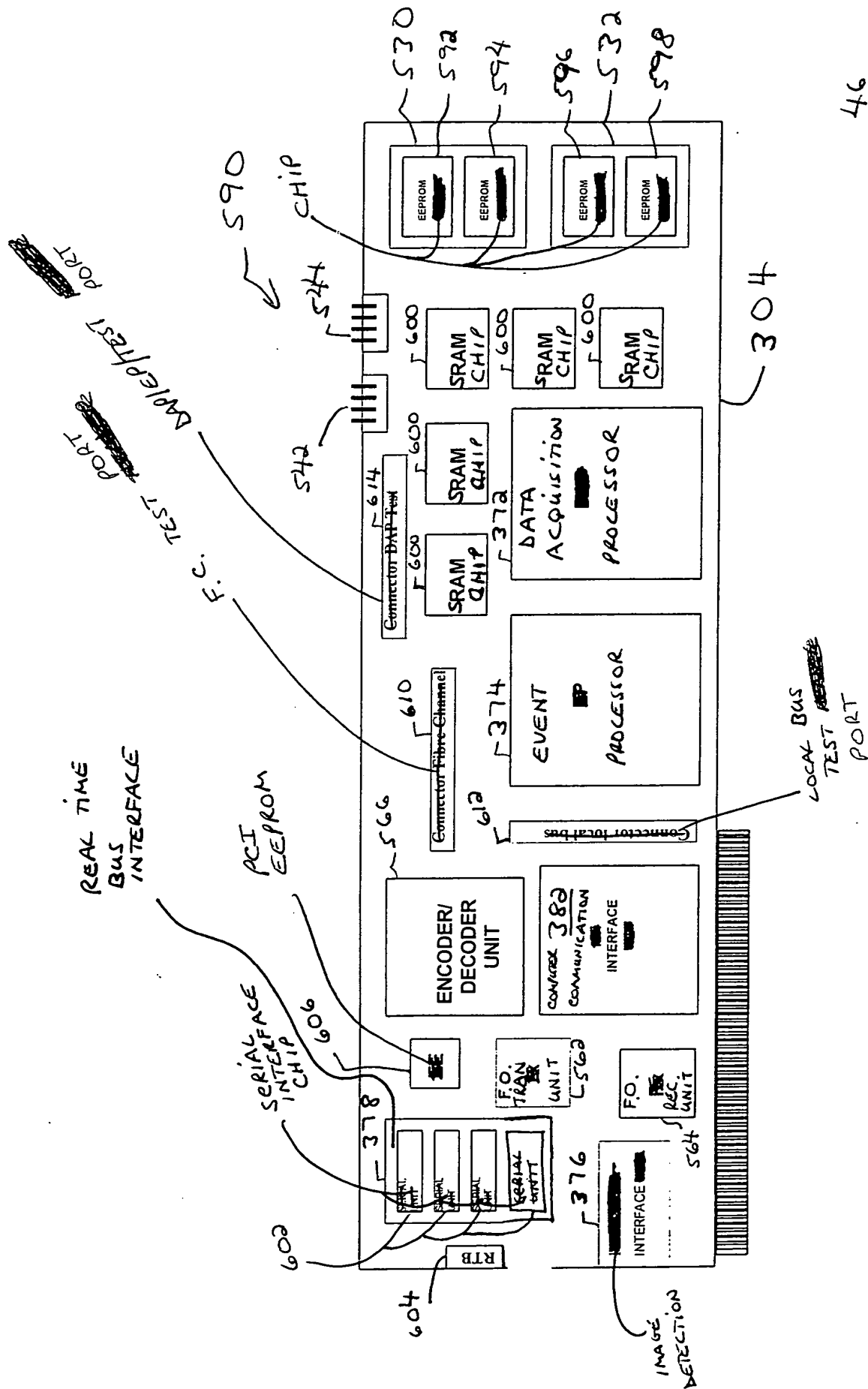
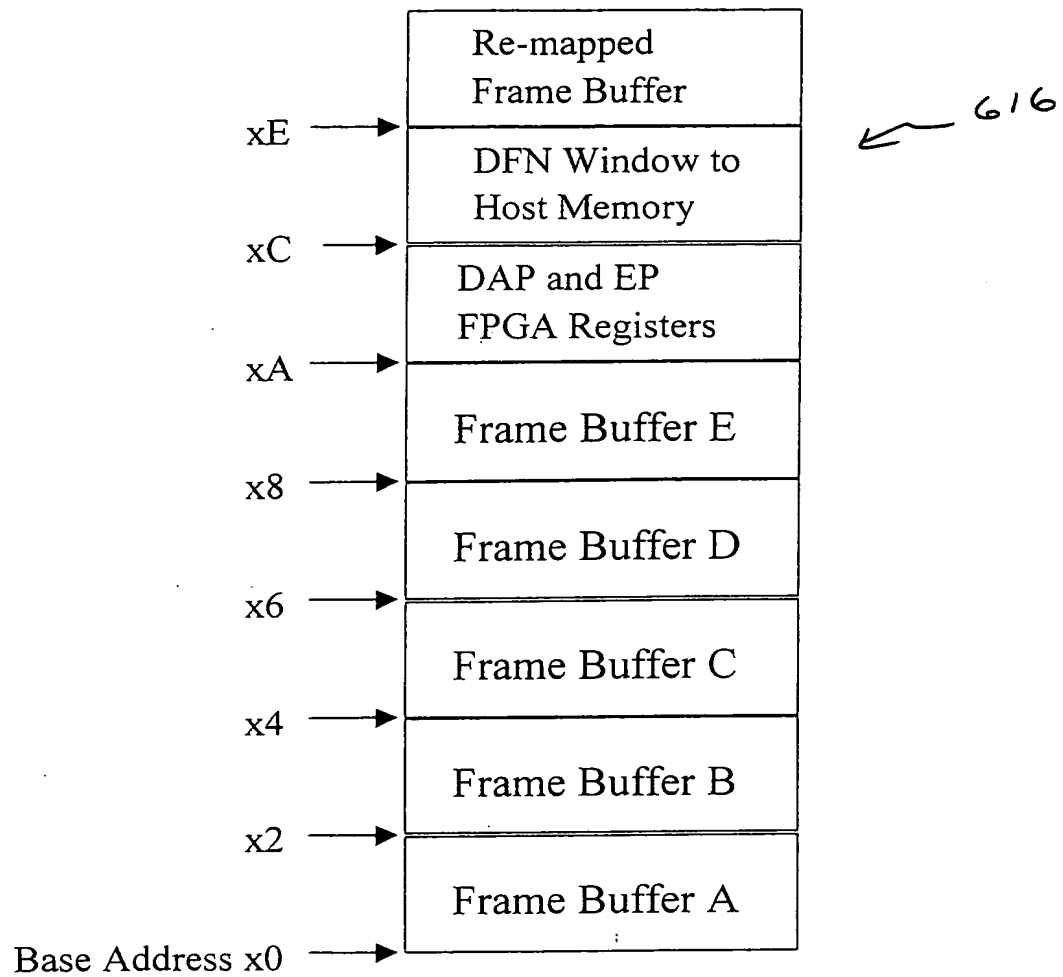


FIG.

46



Mapping of 16 MByte PCI Address Space

FIG. ~~41~~ 47

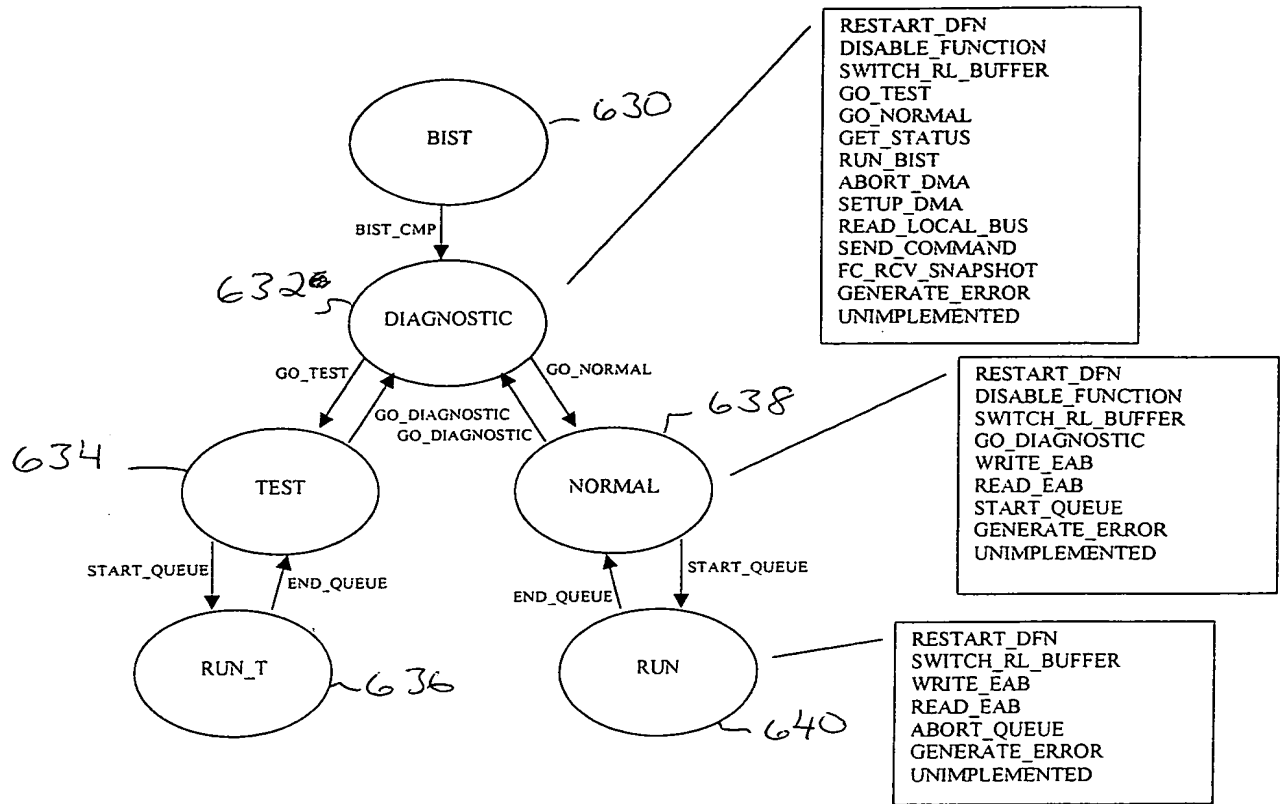


FIG. 48

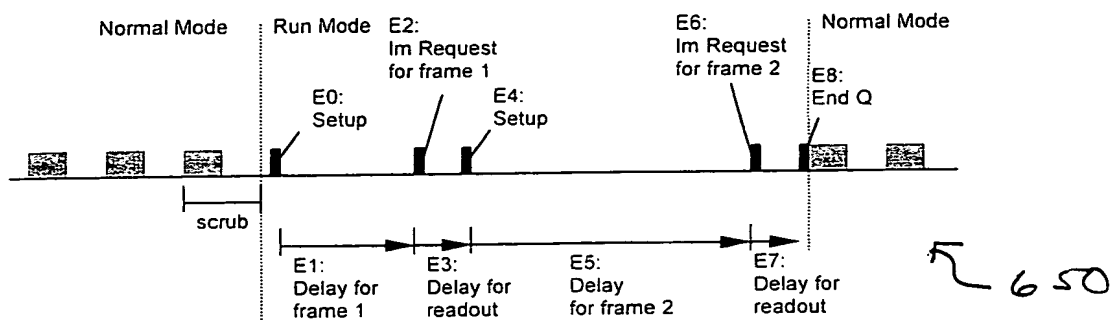


FIG. 49

Event Mnemonic	Event (showing size of arguments)	Op Code (hex)	Data (bytes)	Total (bytes)
Endq	Endq	14	0	1
Delay (I)	Delay (0xff ff ff ff)	10	4	5
Send (command, value)	Send (0xff ff ff ff, 0xff ff ff ff)	04	8	9
LoopKN (K, N)	LoopKN (0xff ff, 0xff)	0C	3	4
LoopKF (K, F)	LoopKF (0xff ff, 0xff ff ff)	0D	5	6
Wait (F)	Wait (0xff ff ff)	09	3	4
Flag (F)	Flag (0xff ff ff)	08	3	4

FIG. 50

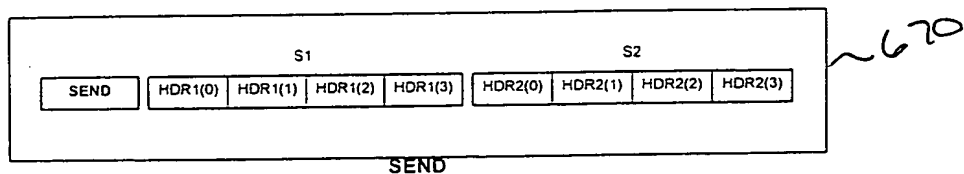


FIG. 51

Error Mnemonic	Description of Error
FC_TIMEOUT	Timeout expired with no ACK detected
FC_BAD_ACK	ACK did not match transmitted command
FC_EXTRA_ACK	Unexpected ACK received
FC_EXTRA_CMD	New Send event while waiting for ACK from previous Send
SIG_DET_N	No input signal power on Fibre Channel (cable disconnected?)
RX_ERROR	Fibre Channel receiver detected bad data (defective chipset?)
WRDSYNCH	Fibre Channel Data link unsynchronized
CRXS(1)	Bad Received CRC detected (Fiber-optic cable problem?)
CRXS(3) and CRXS(2)	Bad order in link state machine (defective chipset?)

← 672

FIG. 52

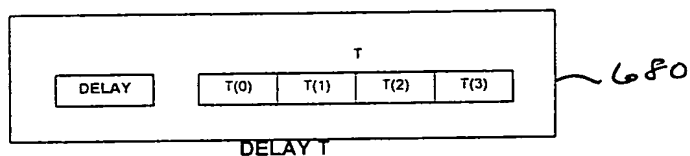


FIG. 53

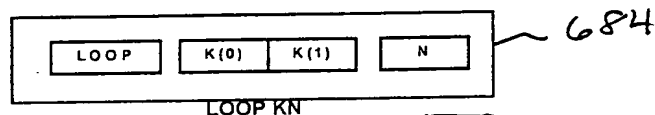
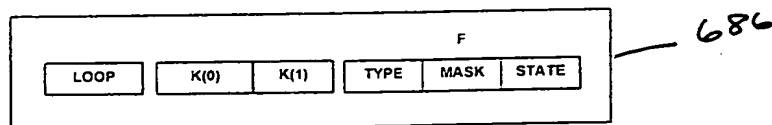


FIG. 54



LOOP KF

FIG. 55

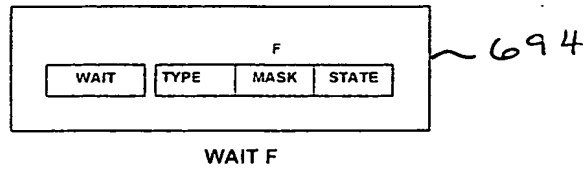


FIG. 56

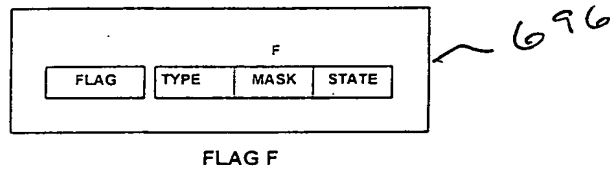


FIG. 57



FIG. 58

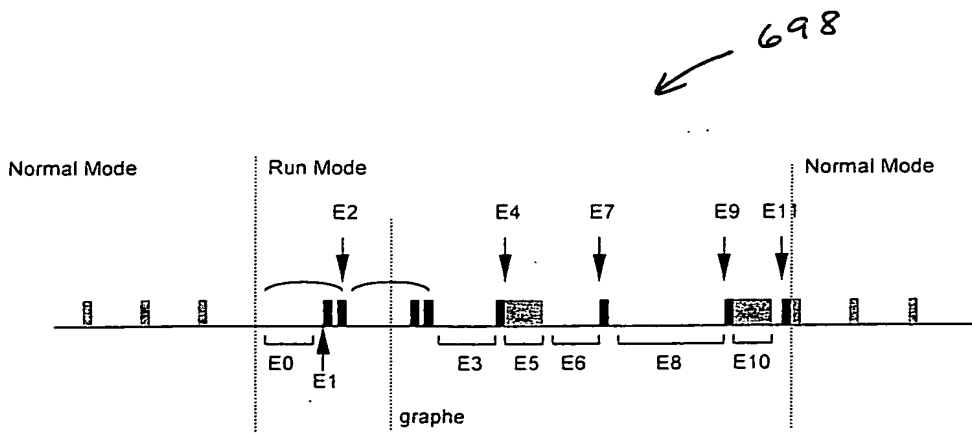


FIG. 59

E11	EndQ
E10	Delay 125 ms
E9	Send Im Request
E8	Delay 500 ms
E7	Flag RT2
E6	Delay 50 ms
E5	Delay 125 ms
E4	Send Im Request
E3	Delay 300 ms
E2	Loop 2, RT1
E1	Send Scrub
E0	Delay 300 ms

Event Queue

FIG. ~~60~~ 60

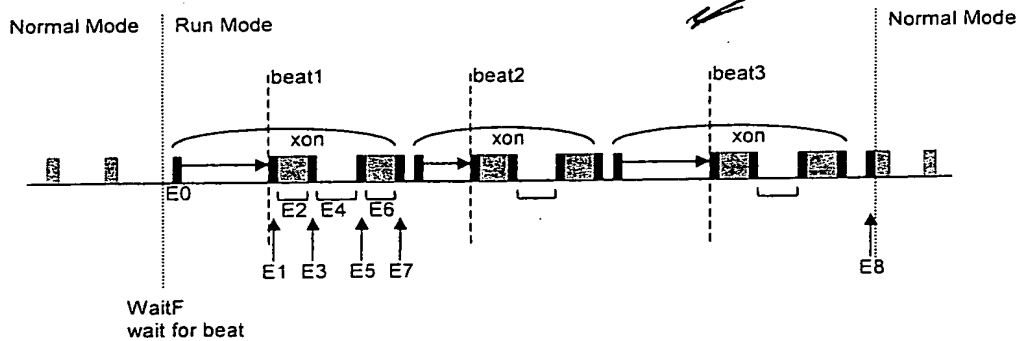
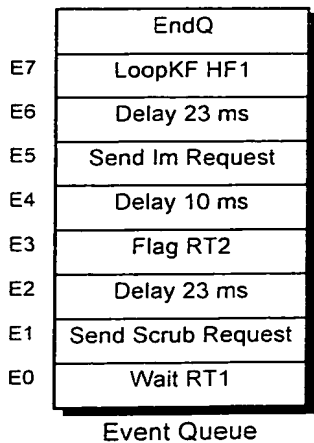


FIG. ~~61~~ 61



← 704

FIG. 62

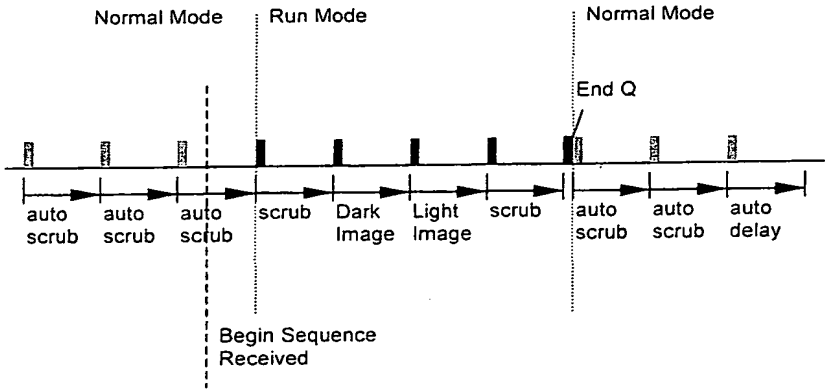


FIG. 63

```
sequence_begin ();

# define qv defaults:
%qv1 = ('delay_qv' => 5000);

# call frame with qv's
frame_type1 (NULL, \%qv1, 1);

sequence_end ();
```

FIG. 64

```
sub frame
{
    $QVf = 'frame';

    %qv = ('delay_qv' => [10000]);
    %qp = ();

    compile_init(@_, \%qp, \%qv, $QVf);

    Delay('delay_qv');

    compile_finit();
}
```

FIG. 65

```
pDFN->DFNChangeQueueVariable
(
    (char *)SymName,      // variable name
    (char *)sndBuf,      // new value
    BufSize,             // num bytes to write
    (ULONG *)&debug     // developer info
);
```

FIG. 66

User Application

```
// load and run the event sequence
pDFN->DFNBeginSequenceNoMappingNoLog
(snum, "d:\\HF.bin");

//assign data to be passed
sndBuf = 25000;

// change the queue variable
pDFN->DFNChangeQueueVariable
(
    (char *)SymName,      // variable name
    (char *)sndBuf,      // new value
    (ULONG)sizeof sndBuf, // num bytes to write
    (ULONG *)&debug     // developer info
);
```

FIG. 67

Perl Script

```
sub frame_type1
{
    $HFfrm = 'frame_type1';

    %qv = ('delay_qv' => [20000] );
    %qp = ();

    $image_cmd = [0x800000, 0x0];

    compile_init(@_, \%qp, \%qv, $HFfrm);

    Send($image_cmd);
    Delay('delay_qv');
    LoopKF(2, 0xAAFF01);

    compile_finit();
}
```

FIG. 68

\\INTSERVER\WORK\11346\todd-DFN cases\drawings\memory management architecture.doc

IMAGE DETECTION SUBSYSTEM

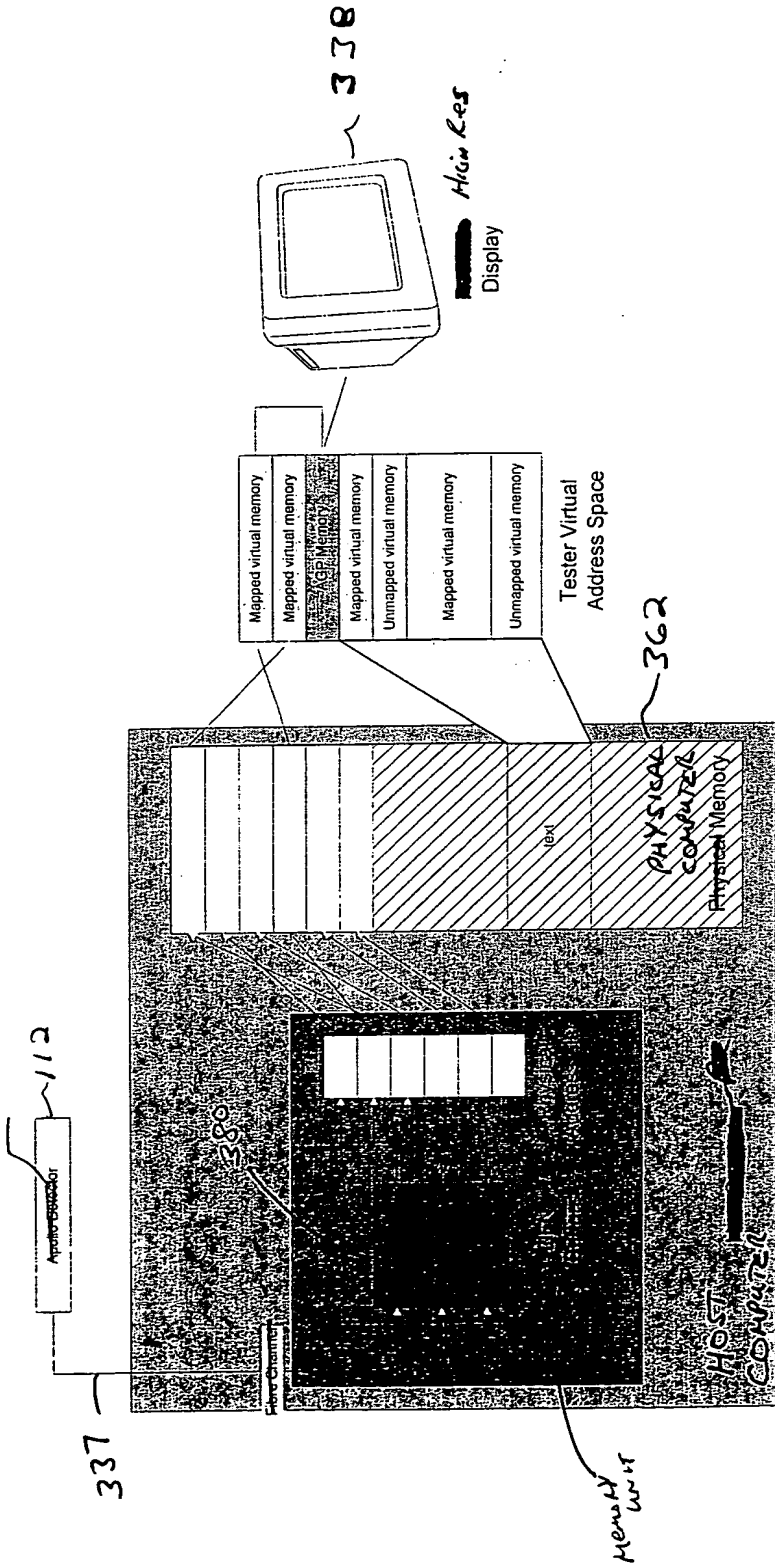
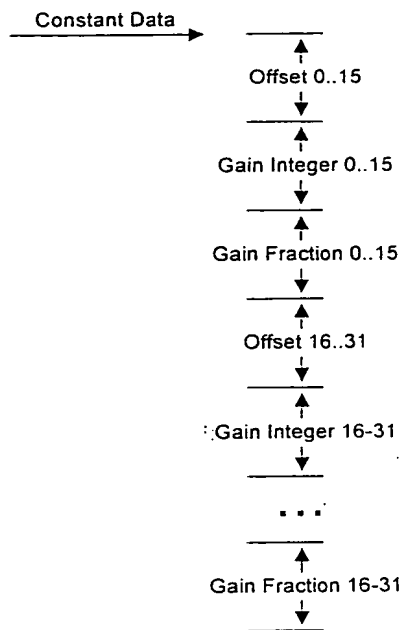


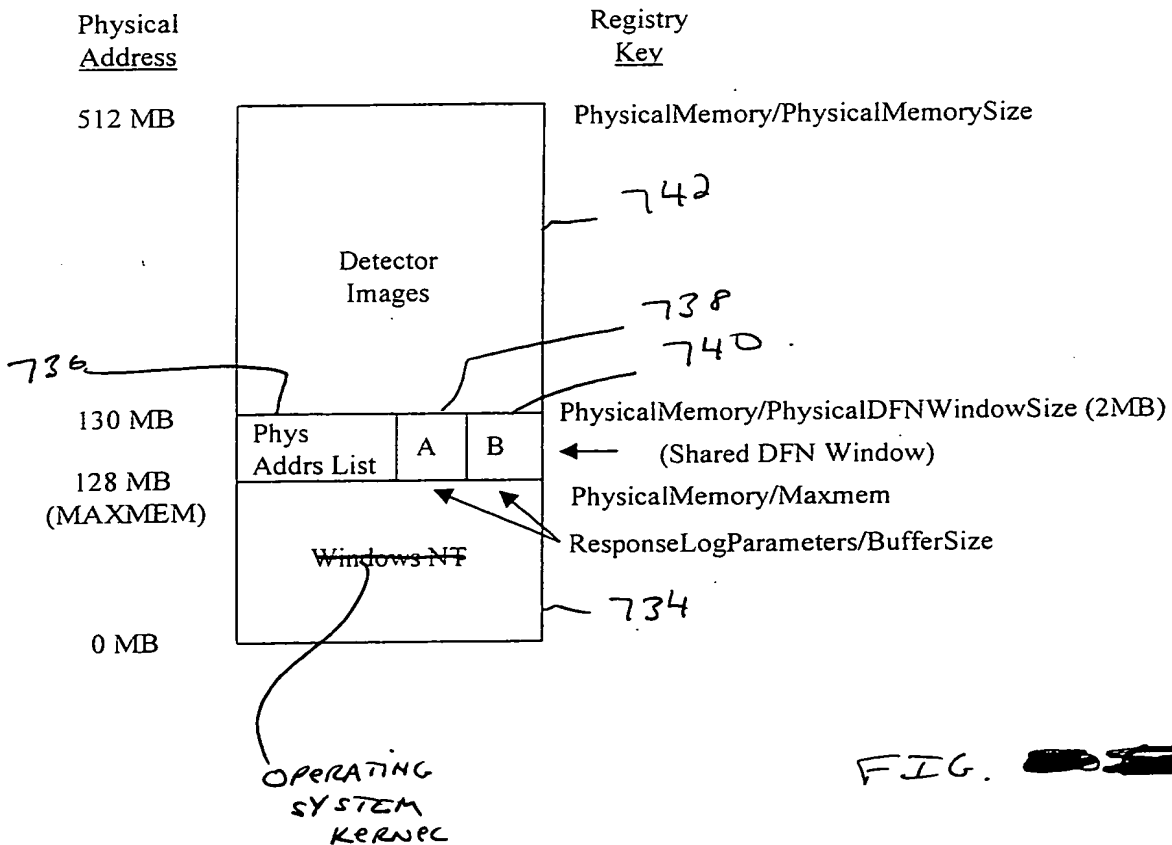
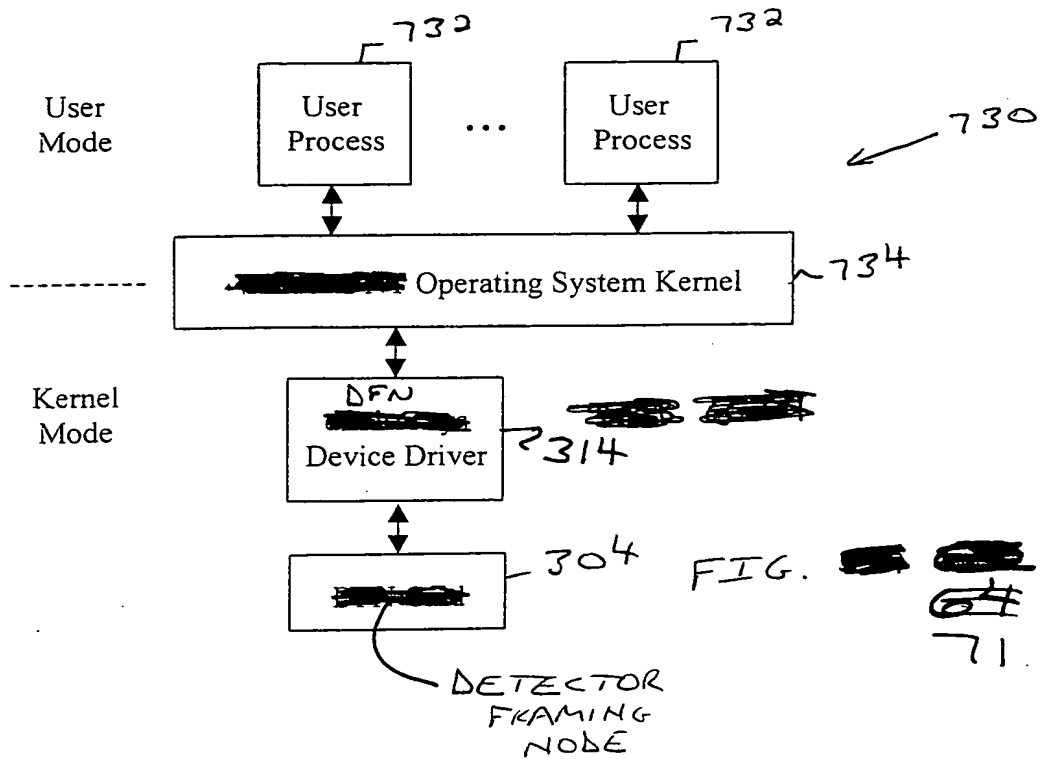
FIG. 69



Constant Memory Format

FIG.

70



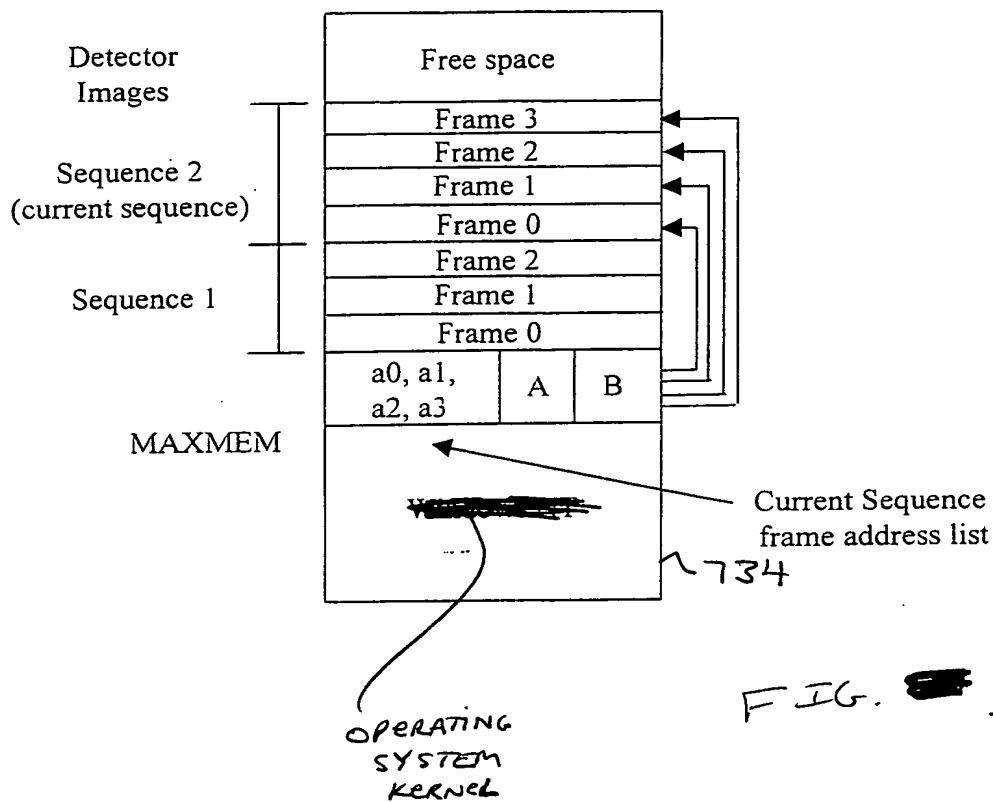
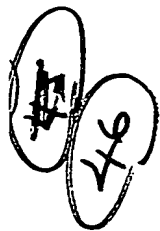


FIG. 73

TOP SECRET